# VERIFICATION OF TRANSLATION

I, Kyozo Omori, translator of 831-9, Ono, Sanda, Hyogo, Japan, hereby declare that I am conversant with the English and Japanese languages and am a competent translator thereof. I further declare that to the best of my knowledge and belief the following is a true and correct translation made by me of Japanese Patent Application No. H11-236724 filed on August 24, 1999.

Date: June 27, 2001

Kyozo Omori

KYOZO OMORI

[DOCUMENT] Patent Application

[OUR REFERENCE NO.] 2022510377

[FILING DATE OF APPLICATION] August 24, 1999

[To] Commissioner, Patent Office

[INTERNATIONAL PATENT CLASSIFICATION] G11B 27/00

G11B 19/02

[INVENTOR]

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Kenji TAGAWA

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Teruto HIROTA

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Tomokazu ISHIKAWA

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Hideki MATSUSHIMA

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Shinji INOUE

[ADDRESS] c/o Matsushita Electric Industrial Co., Ltd.
1006, Kadoma, Kadoma-City, Osaka

[NAME] Masayuki KOZUKA

[APPLICANT]

[CODE NO.] 000005821

[NAME] Matsushita Electric Industrial Co., Ltd.

[PATENT AGENT]

[CODE NO.] 100097445

[PATENT ATTORNEY]

[NAME] Fumio IWAHASHI

[APPOINTED PATENT AGENT]

[CODE NO.] 100103355

[PATENT ATTORNEY]

[NAME] Tomoyasu SAKAGUCHI

[APPOINTED PATENT AGENT]

[CODE NO.] 100109667

[PATENT ATTORNEY]

[NAME] Hiroki NAITO

[CHARGES]

[RECEIPT NO.] 011305

[AMOUNT] ¥21,000

[LIST OF ENCLOSURES]

Specification 1

Drawings 1

Abstract 1

[POWER OF ATTORNEY NO.] 9809938

[PROOF] Unnecessary

[DOCUMENT] Specification

[TITLE OF THE INVENTION] DIGITAL DATA STORAGE MEDIUM, DIGITAL DATA RECORDING APPARATUS, AND DIGITAL DATA REPRODUCING APPARATUS

[CLAIMS]

[CLAIM 1] A storage medium for storing digital data, the storage medium storing:

pieces of presentation data each of which at least includes either audio information or image information; and

pieces of management information each of which corresponds to a piece of presentation data and is used for managing the corresponding piece of presentation data, wherein

the pieces of management information logically manage the pieces of presentation data using (a) frames which are minimum units of the audio information, (b) elements composed of a predetermined number of frames, and (c) blocks composed of consecutive effective elements in the pieces of presentation data, and

each piece of management information includes

information indicating a data length of an ineffective area that is located at the start of a presentation data file,

information indicating an effective data length in the presentation data file,

information indicating a data length between a reference address of the element and the start of the presentation data

file,

information indicating the number of elements in the block,

information indicating the number of frames in the first element of the presentation data file,

information indicating the number of frames in the last element of the presentation data file, and

information indicating the number of frames in elements other than the first and last elements of the presentation data file.

[CLAIM 2] The storage medium of CLAIM 1, wherein

each management information includes

information indicating addresses of the elements in the corresponding piece of presentation data, and

connection information that indicates whether the corresponding piece of presentation data is connected to another piece of presentation data, wherein

the information indicating addresses of the elements has a predetermined data length.

[CLAIM 3] A recording apparatus for recording data onto the storage medium of CLAIM 2, comprising:

a judging means for making a judgement concerning the fixed data length of the information indicating addresses of the

elements; and

a recording means for generating a new piece of management information when the judging means judges that the information indicating addresses of the elements has a data length exceeding the predetermined data length when recording the corresponding piece of presentation data onto the storage medium, and recording the information indicating addresses of the elements into the generated piece of management information.

[CLAIM 4] A reproducing apparatus for reproducing data stored in the storage medium of CLAIM 2, comprising:

a judging means for referring to the connection information in the management information for each piece of presentation data, and judging whether it is necessary to continuously reproduce pieces of presentation data;

an extracting means for extracting appropriate pieces of presentation data when the judging means judges that it is necessary to continuously reproduce pieces of presentation data; and

a reproducing means for decoding and reproducing the extracted pieces of presentation data.

[DETAILED DESCRIPTION OF THE INVENTION]

0001

[FIELD OF THE INVENTION]

The present invention relates to a storage medium which stores digital data containing audio and/or image information in a rewritable state, and to a reproducing apparatus for the storage medium. Particularly, the present invention relates to a storage medium whose recording areas can be used effectively, and to a recording/reproducing apparatus for the storage medium.

0002

[DESCRIPTION OF THE RELATED ART]

The mini disc (MD) has achieved widespread use as a storage medium for storing digital data in a rewritable state. The MD has 140MB of storage capacity that corresponds to approximately 74 minutes of reproduction of compressed digital audio data. One of prevalent styles of using MD is to record several and 10 songs from a music CD to an MD and listens to the songs by reproducing them with a portable machine.

0003

Meanwhile, the music data is stored in MDs as plain texts without encryption. However, copyright owners strongly demand, from the viewpoint of copyright protection, that music data be encrypted before recorded onto MDs.

0004

[THE PROBLEMS THE INVENTION IS GOING TO SOLVE]

One of great problems concerning encrypted recording of music data is in what units the music data should be encrypted. Suppose, for example, all songs to be stored in a storage medium

are encrypted with the same encryption key. In this case, once the key is broken, all the songs are easily decrypted. As understood from this, it is desired to achieve a data structure in which the music data and different keys assigned for the songs are managed.

0005

Another problem concerning encrypted recording of music data is how to simplify the editing of songs. Here, the editing of songs includes a combination (combining a plurality of songs into one) and a division (dividing one song into a plurality of songs).

0006

Suppose, for example, two songs that have been encrypted using different encryption keys are going to be combined into one song with one encryption key. One method for achieving this is to decrypt one of the two songs and encrypt it using the encryption key of the other one. This method, however, is not realistic from the viewpoint of the processing speed. As understood from this, it is desired to achieve a data structure in which one song can be encrypted using a plurality of encryption keys.

0007

Also, it is desired to achieve a data structure in which each editing operation requires a small amount of data (hereinafter referred to as time search table) that is used for achieving a precise time display after a search operation such as fast-forward

or rewinding.

0008

It is therefore an object of the present invention to provide a storage medium having a data structure in which pieces of music data to be encrypted are managed and small amounts of data are required for the editing operations such as division or combination, and provide a recording/reproducing apparatus for recording/reproducing data on the storage medium.

0009

[MEANS TO SOLVE THE PROBLEMS]

As a storage medium for solving the above problems, the present invention provides a storage medium for storing digital data, the storage medium storing: pieces of presentation data each of which at least includes either audio information or image information; and pieces of management information each of which corresponds to a piece of presentation data and is used for managing the corresponding piece of presentation data, wherein the pieces of management information logically manage the pieces of presentation data using (a) frames which are minimum units of the audio information, (b) elements composed of a predetermined number of frames, and (c) blocks composed of consecutive effective elements in the pieces of presentation data, and each piece of management information includes information indicating a data length of an ineffective area that is located at the start of a presentation data file, information indicating an effective data

length in the presentation data file, information indicating a data length between a reference address of the element and the start of the presentation data file, information indicating the number of elements in the block, information indicating the number of frames in the first element of the presentation data file, information indicating the number of frames in the last element of the presentation data file, and information indicating the number of frames in elements other than the first and last elements of the presentation data file.

0010

In the above storage medium, each management information may include information indicating addresses of the elements in the corresponding piece of presentation data, and connection information that indicates whether the corresponding piece of presentation data is connected to another piece of presentation data, wherein the information indicating addresses of the elements has a predetermined data length.

0011

The present invention also provides a recording apparatus for recording data onto the storage medium of CLAIM 2, comprising: a judging means for making a judgement concerning the fixed data length of the information indicating addresses of the elements; and a recording means for generating a new piece of management information when the judging means judges that the information indicating addresses of the elements has a data length exceeding

the predetermined data length when recording the corresponding piece of presentation data onto the storage medium, and recording the information indicating addresses of the elements into the generated piece of management information.

0012

The present invention also provides a reproducing apparatus for reproducing data stored in the storage medium of CLAIM 2, comprising: a judging means for referring to the connection information in the management information for each piece of presentation data, and judging whether it is necessary to continuously reproduce pieces of presentation data; an extracting means for extracting appropriate pieces of presentation data when the judging means judges that it is necessary to continuously reproduce pieces of presentation data; and a reproducing means for decoding and reproducing the extracted pieces of presentation data.

0013

[EMBODIMENTS OF THE INVENTION]

The following describes with reference to the drawings the construction of a storage medium as an embodiment of the present invention.

0014

In the present embodiment, music data is used as the object of the processes. However, not limited to the music data, image data, text data, or a combination of these types of data may be

used as the object of the processes.

0015

Embodiment 1

The data structure of the storage medium, a semiconductor memory of the present invention will be described. The semiconductor memory of the present invention (hereinafter referred to as media card) is, as is the case with DVD (Digital Video Disc), composed of a physical layer, a file system layer, and an application layer. Each layer will be described.

0016

FIG. 1 shows the shape of the media card. As shown in FIG. 1, the media card is approximately 30.0mm long, 23.0mm wide, 2.0mm thick. The media card is a readable/writable storage medium and has the sector structure. Each sector has the capacity of 512 bytes. For example, in the case of a 64MB-type media card, when the memory capacity is 65,536,000 bytes, the number of effective sectors is 138,000. Note that in reality, the total amount of storage areas available for the user is smaller than the capacity since alternative sectors are prepared for errors.

0017

FIG. 2 shows the construction of the areas in the media card. As shown in FIG. 2, the media card has a special area, an authentication area, and a user area. Of these areas, the special area and the authentication area are used for the copyright

protection.

0018

The special area is a read-only area, and stores media IDs that have values unique to media.

0019

The authentication area becomes available for reading or writing only when a mutual authentication between a personal computer connected to the media card and a player succeeds.

0020

The user area, as is the case with the flash ATA card or the compact flash, can be read or written by a typical application.

0021

The data protected by copyright is encrypted using a key (file key) generated from a media ID, a random number or the like, and is stored in the user area. The key is encrypted using a secret key to be an encrypted key, and is stored in the authentication area.

0022

As described above, the media card has a function to prevent illegal copying of data or the like since the data protected by copyright is encrypted before it is stored in the media card.

0023

Now, the file system layer will be described.

0024

The file system used in the media card is FAT (File Allocation Table) file system (ISO/IEC 9293). The file system supports two types, FAT12 and FAT 16. The authentication area and the user area of the media card are formatted as FAT file systems.

0025

FIG. 3 shows the construction of the file system in the media card. The file system is composed of a partition boot sector, a file allocation table, a root directory entry, and a data area. The authentication area and the user area have the same construction. Each element of the construction will be described.

0026

The partition boot sector is a sector that is read when the system is activated.

0027

The file allocation table supports two types of file systems: the FAT-12 file system for 12-bit FAT; and the FAT-16 file system for 16-bit FAT. The FAT structure conforms to ISO/IEC 9293. The total number of clusters that decides FAT 12 and FAT 16 can be determined by a parameter of the physical layer.

0028

Each element of the file system is placed at a boundary that is determined by a parameter of the physical layer. This

prevents the saving process of the flash memory in the media card from occurring. For example, the starting address of the data area is placed at a boundary that is determined by a parameter of the physical layer. The cluster size is set to be the same value as the starting address. This prevents the saving process from being executed when the data area is accessed.

0029

FIG. 4 shows the construction of the file system which is composed of directories and files. The user area contains different encryption keys for encrypted files. This is because even if an encryption key for a certain file is broken, it does not affect the other encrypted files. The encryption keys used for the encrypted files are stored in an encryption key storing file corresponding to the authentication area. Also, the relationships between the encrypted files and the encryption keys are determined in accordance with the following rules.

(1) Allocated to the same directory name as that of the data area.

(2) Has a file name which is a combination of (a) the first three characters of the name of an encrypted file in the data area and (b) an extension.

(3) The extensions are ".KEY" and ".BUP".

(4) The number in the name of an encrypted file in the data area corresponds to the number of File Key Entry.

0030

With the above rules, an encryption key is uniquely determined for an encrypted file.

0031

Up to now, the file system of the media card of the present invention has been described.

0032

Now, the presentation data unit will be described.

0033

The presention data of the present invention has the construction shown in FIG. 12. That is to say, the presention data is composed of: (a) audio object (hereinafter referred to as AOB) which is operated by the navigation data, (b) image object (IOB), (c) a time search map (TMSRT) for managing reproduction time of the AOB, and a block information table (BIT). Each of these components will be described.

0034

The construction of AOB will be described with reference to FIG. 14.

0035

The AOB is managed by TKI included in navigation data. The AOB is roughly composed of three layers constituting a hierarchy. The lowest layer is a minimum unit of AOB, AOB_FRAME.

0036

A layer higher than the AOB_FRAME is AOB_ELEMENT which is a sequence of AOB_FRAMEs. The number of AOB_FRAMEs per AOB_ELEMENT

is shown in FIG. 13. However, the number of AOB_FRAMEs in each of the first and last AOB_ELEMENTs of an AOB may be lower than that shown in FIG. 13 when some editing operation (e.g., division) is performed on the AOB. The AOB_ELEMENT is managed by TMSRT which will be described later.

0037

A layer higher than the AOB_ELEMENT is AOB_BLOCK which is an area storing a sequence of effective AOB_ELEMENTs in an AOB. One AOB file contains one AOB_BLOCK.

0038

Now, the type of AOB will be described. Data that is treated as AOB is defined in MPEG2-AAC {Low Complexity Profile}. For MPEG2-AAC, refer to ISO/IEC 13818-7: 1997(E) Information technology --- Generic Coding of moving pictures and associated audio information --- Part 7 Advanced Audio Coding (AAC).

0039

The stream format for MPEG2-AAC is ADTS (Audio Data Transport Stream).

0040

MPEG2-AAC for media cards specifies the parameters written in ISO/IEC13818-7 as shown in FIG. 15.

0041

The parameters other than sampling_frequency_index and channel_configuration are specified in accordance with LC-profile defined in ISO/IEC 13838-7.

0042

Now, IOB will be described. The present media card can output various types of image information such as still pictures in synchronization with the reproduction of AOBs. Such image information is called "IOB". IOBs are encoded in, for example, the JPEG (Joint Photographic Experts Group) format before they are recorded. One IOB file stores one IOB.

0043

The start of the IOB file contains the information shown in FIG. 16.

0044

Here, "IOB_ID" indicates a magic number of the IOB file, and its value is "EEE".

0045

"IOB_ATR" is a flag indicating that the IOB file does not contain an IOB as a substance, but refers to another file. FIG. 17 shows the details of IOB_ATR.

0046

As described above, it is possible to reduce the capacity of the media card by allowing IOB files to refer to other files without storing a substance.

0047

"IOB_SZ" indicates the data length of the IOB.

0048

Now, the time search table (TMSRT) will be described.

0049

"TMSRT" is information indicating the position of "AOB_FRAME" in the AOB file, and is contained in "TKI".

0050

The "TMSRT_entry", a component of TMSRT, indicates the starting address for each AOB_ELEMENT to the AOB files of the initial state recorded on the media card.

0051

FIG. 18 shows an example of TMSRT when TMSRT_entry is obtained per 96 frames.

0052

Now, the block information table (BIT) will be described.

0053

The BIT is used to manage AOB_BLOCKs in AOBs, and is composed of the following entries.

0054

(1) Data_Offset: the size of the invalid area at the start of the AOB file.

(2) SZ_Data: the size of AOB_Block.

(3) TMSRT_Offset: the offset from the reference address of TMSRT to the start of the AOB file. If an AOB is divided, the address of each entry of the divided TMSRT is not rewritten. TMSRT_Offset, therefore, a reference address value which is subtracted to calculate the address of an actual TMSRT entry from the AOB

start.

0055

(4) TMSRT_Ns: the number of entries of TMSRT in the current AOB_Block.

(5) FNs_1st_TMSRTE: the number of frames in the first TMSRT entry.

(6) FNs_Last_TMSRTE: the number of frames in the last TMSRT entry.

(7) FNs_Middle_TMSRTE: the number of frames in the middle TMSRT entry.

FIG. 19 shows the entry relationships between AOB and BIT.

0056

The reproduction operation of AOB will be described.

When a player reproduces tracks, first it selects a playlist included in the navigation data.

0057

The selection of playlist will be described.

0058

The Playlist Manager (PLMG) first contains the default playlist management information (DPLI), then up to 99 pieces of playlist management information (PLI). The playlists are numbered as 1, 2, ... 99 in the order of description in the PLMG.

0059

Ordinarily, DPLI is read first. However, when an automatic

call of a playlist is specified in PLMG_TK_PL in PLMG, the Playlist Manager Information (PLI) is called. There are two types of automatic calls: Bookmark and Resume.

0060

FIG. 20 is a schematic representation of the selection of a playlist.

0061

After a playlist is selected, a song can be reproduced. Each DPLI or PLI contains pieces of track management information (TKI) reference information (DPL_TK_SRP and PL_TK_SRP) in the order of reproduction.

0062

In DPLIs and PLIs, the DPL_TK_SRPs and PL_TK_SRPs are treated as songs 1, 2, ... 99 (in the default playlist, up to 999) in the order of description.

0063

FIG. 21 is a schematic representation concerning the song reproduction order.

0064

Note that in DPLI, TKIs are treated as songs only when the flag of DPL_TK_SRP indicates the start of a song.

0065

TKI reproduces music using AOB and TMSRT, and displays IOB in accordance with the information written in Display Mode. FIG. 22 relates to TKI reference information.

0066

The ordinary reproduction is performed in the following procedure using the selected AOB and TMSRT.

(1) Obtain Data_Offset, SZ_DATA, and Fns_1st_TMSRT from BIT.

(2) Enter AOB_FRAME as indicated by Fns_1st_TMSRTE, from Data_Offset into the decoder.

(3) Increment the reproduction time by (Fns_1st_TMSSRTE*reproduction period of 1 AOB_FRAME).

(4) After (3), increment the reproduction time by the reproduction period of 1 AOB_FRAME each time AOB_FRAME is entered into the decoder.

(5) Continue the operation of (4) above until the number of pieces of data entered into the decoder becomes SZ_DATA and the number of frames entered after TMSRT_entry becomes Fns_Last_TMSRTE.

0067

The intermittent reproduction is performed in the following procedure using the selected AOB and TMSRT. For example, 200 milliseconds of reproduction is performed every 2 seconds in the intermittent reproduction.

(1) Obtain Data_Offset, SZ_DATA, and Fns_1st_TMSRTE, TMSRT_Offset from BIT.

(2) Enter AOB_FRAME equivalent to the intermittent reproduction time into the decoder from the position reached by skipping an amount indicated by Data_Offset from the start of AOB.

(3) Increment the reproduction time by a value corresponding to the

intermittent reproduction time.

(4) Detect where in the TMSRT entry corresponds to the frame position after the increment.

(5) Detect, from TMSRT, TMSRT_entry that includes the position obtained by incrementing by the number of AOB_FRAMEs corresponding to the intermittent interval.

(6) Check AOB_ELEMENT in the TMSRT_entry to detect the position of the frame in AOB_ELEMENT.

(7) Enter AOB_FRAME corresponding to the intermittent reproduction time to the decoder, from the detected frame position.

(8) Return to (3) and continue the operation until the end of the intermittent reproduction.

0068

The recording operation will be described.

(1) One TMSRT_entry is added per the number of frames constituting AOB_ELEMENT (FNs_Middle_TMSRTENUM_TMSRTENUM_TMSRTENT_WIDTH).

(2) Since the size of TMSRT is 1 KB, TMSRT can store 256 entries.

(3) One TMSRT_entry is added per time corresponding to the number of frames constituting AOB_ELEMENT. When the total number of TMSRT_entry becomes 256, a new TKI(TMSRT) is generated, and a copy of a former BIT is made. One TMSRT_entry is added to the newly generated TMSRT per time corresponding to the number of frames constituting AOB_ELEMENT.

(4) Repeat (3) until the end of input signals.

0069

Now, AOB division will be described. In the present embodiment, the division indicates that, for example, a piece of music data is divided in units of AOB_FRAMEs.

0070

The division will be explained using an example in which an AOB is divided into two at the address Q between the $k^{th}$ TMSRT_entry (TMSRT_entry #k) and the $(k+1)^{th}$ TMSRT_entry (TMSRT_entry #k+1) in the AOB. The first and second parts of the AOB before the division will be respectively referred to as Song 1 and Song 2 after the division, for the sake of convenience.

0071

One AOB is divided into two AOBs that each include one AOB_BLOCK. FIG. 23 shows the case where AOB_BLOCK after division includes a plurality of TMSRT_entry. In the present example, the AOB is divided at the $p^{th}$ frame including the address Q in AOB_ELEMENT.

0072

The TMSRT and BIT change as follows in by the division shown in FIG. 23.

0073

Firstly, the TMSRT changes as follows.

0074

The first TMSRT includes the first to the $K^{th}$ entry in TMSRT of AOB before division.

0075

The second TMSRT includes the $(k+1)^{th}$ entry to the last entry (TMSRT_entry #n) in TMSRT of AOB before division. Furthermore, one TMSRT_entry is added to the start.

0076

FIG. 24 shows an example of changing TMSRT by division.

0077

BIT is as follows.

0078

In the first BIT, SZ_DATA is changed to the data length up to the division point Q, TMSRTE_Ns to (k+1), and Fns_Last_TMSRTE to p frame.

0079

In the second BIT, Data_Offset is changed to Q, SZ_DATA to the data length up to the division point Q, TMSRTE_Ns to (n-k) (this means that K+1 for the first song and n-k are added up to N+1 for the original AOB), MSRT_Offset to the cluster position of the AOB before division including the division point, FNs_1st_TMSRTE to 96-p frame, and FNs_Last_TMSRTE to 50 (the same as the original AOB).

0080

When TMSRT_entry is not included in one of AOB_BLOCKs after division, FNs_1st_TMSRTE in the corresponding BIT becomes 0, and TMSRTE_Ns also becomes 0.

FIG. 25 shows an example of how BIT changes by division.

0081

FIG. 26 shows a system model when AOB is decoded.

0082

An AOB file is input to BitstreamReader. An AOB_BLOCK is taken out from an AOB in accordance with the information written in the BIT. The AOB_BLOCK is input to the AudioBuffer. AOB_BLOCKs accumulated in the AudioBuffer are input to the Deformatter. ADTS headers are detected, and at the same time, AOB_ELEMENTs are taken out. The total number of the detected ADTS headers is managed by the HeaderParser, and is sent to the navigation layer as necessary. The AOB_ELEMENTs taken out from the Deformatter are divided into AOB_FRAMEs in accordance with the number of ADTS headers managed by AOB_EHeaderParser. The AOB_FRAMEs are entered into the AudioDecoder. The AudioDecoder decodes the entered AOB_FRAMEs one by one and obtains PCM data.

0083

Up to now, the presentation data has been described.

0084

Now, the navigation data will be described.

0085

The navigation data relates to the attributes and reproduction control of the presentation data. As shown in FIGs. 27, 28, and 29, the navigation data is composed of three logical components: Playlist Manager (PLMG), Track Manager (TKMG), and IOB Manager (IOBMG). PLMG includes Default Playlist Information (DPLI)

and Playlist Information (PLI). PLMG contains information relating to playlist, and also contains information relating to texts and still pictures for the playlist. TKMG includes Track Information (TKI) and stores reference information and management information of each song. IOBMG includes IOB Count Information (IOBCI) and manages whether each IOB file is referred to by playlists or TKIs.

0086

The data size of each component will be described. As shown in FIG. 27, Playlist Manager Information (PLMGI) and Default Playlist Information (DPLI) have a fixed length of 512 bytes in total. Playlist Information (PLI) has also a fixed length of 512 bytes. As shown in FIG. 28, in Track Manager (TKMG), each Track Information (TKI) has a fixed length of 1536 bytes. As shown in FIG. 29, IOB Manager (IOBMG) has a fixed length of 2048 bytes.

0087

Each component will be described in detail.

0088

The construction of Playlist Manager (PLMG) will be described.

0089

In the present embodiment, the playlist is information that defines the reproduction order of songs. The playlist is classified into two types: a default playlist for managing all tracks (songs) stored in the medium; and a playlist that can be

defined by the user.

0090

Playlist Manager (PLMG) contains information relating to playlists. As shown in FIG. 27, Playlist Manager (PLMG) first contains Playlist Manager Information (PLMGI) for managing playlists stored in the medium, then contains Default Playlist Information (DPLI) for managing all songs stored in the medium, then contains as many pieces of Playlist Information (PLI), which can be defined by the user, as there are playlists. The maximum number of playlists is 99.

0091

Playlist Manager Information (PLMGI) and Default Playlist Information (DPLI) have a fixed length of 512 bytes in total. Playlist Information (PLI) also has a fixed length of 512 bytes.

0092

Each component will be described in detail.

0093

As shown in FIG. 30, Playlist Manager Information (PLMGI) contains PLMG size, the number of playlists stored in the medium, and auto-play playlist information. These pieces of information will be described in detail.

0094

PLMG_ID contains ID used for identifying PLMGI uniquely.

0095

SDA_ID contains a character sequence of "SD-AUDIO" written in ISO646 character codes and indicating that the data conforms to the SD-AUDIO standard.

0096

VERN contains a version number of the SD-AUDIO standard. As shown in FIG. 31, the bits from bit b7 to bit b0 contain information indicating the version number. For example, the version 0.9 is represented as "09h", and the version 1.0 as "10h". The bits from bit b15 to bit b8 are reserved for a future extension.

0097

PLMG_PL_Ns contains the number of playlists treated by PLMG, that is to say, the number of playlists recorded on the medium.

0098

PLMG_AP_PL indicates a playlist automatically read out when the player is activated and also indicates song numbers of the playlist. As shown in FIG. 32, the bits from b7 to b0 indicate one of "1" to "99" indicating a playlist to be automatically read out. This number corresponds to the Playlist Information (PLI) number which will be described later. The default playlist is specified by "0". The bits from b25 to b16 indicate a song number. This number corresponds to the Track Information (TKI) number which will be described later. The bits from b31 to b26 and bits from b15 to b8 are reserved for a future extension.

0099

PLMG_RSM_PL indicates a playlist that was reproduced most recently and also indicates song numbers of the playlist. As shown in FIG. 32, the bits from b7 to b0 indicate one of "1" to "99" indicating a playlist to be automatically read out. This number corresponds to the Playlist Information (PLI) number which will be described later. The default playlist is specified by "0". The bits from b25 to b16 indicate a song number. This number corresponds to the Track Information (TKI) number which will be described later. The bits from b31 to b26 and bits from b15 to b8 are reserved for a future extension.

0100

PLMG_APP_ATR contains SD-CARD application category ID. "01h" represents audio, "02h" game, "03h" video, "04h" book, "05h" karaoke, and "06h" reading book.

0101

Using the application category ID, karaoke can be achieved as follows, for example. When the contents data is karaoke, the right channel is used for recording music, and the left channel for recording audio. The reproduction apparatus outputs only the right channel to both the right and left channels.

0102

PLMG_FCA is reserved for a future extension of the SD-CAED.

0103

TKI_Ns contains an integer indicating the number of TKIs. The maximum number of TKIs is 999.

0104

Up to now, Playlist Manager Information (PLMGI) has been described.

0105

Now, Default Playlist Information (DPLI) will be described.

0106

Default Playlist Information (DPLI) manages all songs in SD-Audio. As shown in FIG. 33, DPLI first contains Default Playlist General Information (DPLGI), then contains Default Playlist Track Search Pointer (DPL_TK_SRP). The medium only stores Default Playlist Information (DPLI). The medium has 999 tracks. As a result, Default Playlist Information (DPLI) can manage 999 songs at the maximum.

0107

Each component will be detailed.

0108

As shown in FIG. 34, Default Playlist General Information (DPLGI) contains the number of songs that are referred to by the default playlist (equal to the number of songs stored in the medium), the total reproduction time of the songs, text information, and still picture information reproduced by the

default playlist.

0109

DPLI_ID contains an ID used for identifying DPLI uniquely.

0110

DPLI_TK_Ns contains the number of songs that are referred to by the default playlist. The maximum number is 999.

0111

DPLI_PB_TM contains the total reproduction time of the all songs that are referred to by the default playlist, in units of milliseconds.

0112

DPLI_APP_ATR contains an application attribute for default playlist.

0113

DPLI_FCA is reserved for a future extension of the default playlist.

0114

Now, the text information of the default playlist will be described. The default playlist can have unique text information. The text information may be divided into two types according to Character set code. Also, the text information of the default playlist can be used as information for identifying media. The following is a description of the two types of text information.

0115

DPLI_PLTI1_ATR contains text attribute information of default playlist. As shown in FIG. 35, the bits from b7 to b0 contain Character set code. "00h" represents ISO646 (ASCII), "01h" JISX0201 (ASCII + Kana (a Japanese set of Characters)), and "02h" ISO 8859-1. The bits from b15 to b8 are reserved for a future extension.

0116

DPLI_PLTI2_ATR contains text attribute information of the default playlist. As shown in FIG. 35, the bits from b7 to b0 contain Character set code. "81h" represents Music Shift JIS Kanji. The bits from b15 to b8 are reserved for a future extension.

0117

DPLI_PLTI contains the text information of the default playlist. When there is no text, the text information is written as "0". When there is a text and the space prepared for the text information is not used entirely, the remaining space is filled with "0".

0118

When there are two texts having different character codes, the text information of the first text corresponding to DPLI_PLTI1_ATR is first written, the end code "0x00" is written, then the text information of the second text corresponding to DPLI_PLTI2_ATR is written. In this case also, when the space prepared for the text information is not used entirely, the remaining space is filled with "0".

0119

DPLI_IOB_SRP contains the still picture (IOB) numbers of still pictures referred to by the default playlist and contains their attributes. 60 numbers are written there. As shown in FIG. 36, the bits from b25 to b16 indicate an IOB number. That is to say, the bits indicate one of 1 to 999 corresponding to an IOB file. When no still picture is referred to, "0" is written. The bits from b15 to b14 indicate a display timing mode. "00h" represents the slide show mode, "01h" the browsable mode. It should be noted here that in the slide show mode, the video synchronizes with the audio and the skip reproduction of only video is not possible. In the browsable mode, the video does not synchronize with the audio and the skip reproduction of only video is possible. The bits from b13 to b12 indicate the display order mode. "00b" represents a sequential display, "01b" a random display, and "10b" a shuffle display. The bits from b11 to b8 indicate the image size. "0000b" represents 160*120, "0001b" 320*240, "0010b" 640*480, "0011b" 800*600, "0100b" 1024*768, and "0101b" 1280*1024. The bits from b7 to b4 indicate the number of colors in still pictures. "0000b" represents 24 bits, "0001b" 16 bits, and "0010b" 8 bits. The bits from b3 to b0 indicate the image coding mode. "0000b" represents JPEG (Joint Photograph Expert Group). The bits from b31 to b26 are reserved for a future extension.

0120

Up to now, the default playlist general information (DPLGI)

has been described.

0121

Now, Default Playlist Track Search Pointer (DPL_TK_SRP) will be described. Default Playlist Track Search Pointer (DPL_TK_SRP) contains TKI reference information. Also, the description order of Default Playlist Track Search Pointer (DPL_TK_SRP) indicates a reproduction order. During the reproduction process, a TKI to be reproduced is specified in accordance with the reference information. As a rule, the reproduction order is equivalent to the order in which the tracks are stored into the medium. A new track is added to the end of the sequence.

0122

The number of prepared DPL_TK_SRPs are 999. If there is no TKI to be referred to, DPL_TK_SRPs are filled with "0".

0123

As shown in FIG. 37, the bits from b12 to b10 of DPL_TK_ATR indicate whether the TKI of a reference target is used. "000b" represents the case where the TKI is used, and one song is included in one TKI. "001b" represents the case where the TKI is used, one song is composed of a plurality of TKIs, and the TKI is the first one of the plurality of TKIs. "010b" represents the case where the TKI is used, one song is composed of a plurality of TKIs, and the TKI is a middle one of the plurality of TKIs. "011b" represents the case where the TKI is used, one song is composed of a plurality of

TKIs, and the TKI is the last one of the plurality of TKIs. "100b" represents the case where the TKI is not used and space for TKI has been allocated, that is to say, it is a deleted TKI. "101b" represents the case where the TKI is not used and space for TKI has not been allocated, that is to say, it is TKI in the initial state. The bits from b9 to b0 of DPL_TKN indicate a TKI number. This enables the reference target TKI to be specified. The bits from b15 to b13 are reserved for a future extension.

0124

Up to now, Default Playlist Track Search Pointer (DPL_TK_SRP) and Default Playlist Information (DPLI) have been described.

0125

Now, Playlist Information (PLI) will be described.

0126

The playlist can be edited by the user and can define the reproduction order of up to 99 tracks. The management information of the playlist is written in Playlist Information (PLI). This enables the user to select any songs stored in the medium and define the reproduction order of the selected songs.

0127

As shown in FIG. 38, PLI first contains Playlist General Information (PLGI), then Playlist Track Search Pointer (PL_TK_SRP). The Playlist General Information (PLGI) manages the whole playlist. The Playlist Track Search Pointer (PL_TK_SRP) contains track

reference information. The number of playlists is 99 at the maximum. Each playlist can manage 99 songs at the maximum.

0128

Each component will be detailed.

0129

First, the Playlist General Information (PLGI) will be described.

0130

As shown in FIG. 39, Playlist General Information (PLGI) contains the number of songs that are referred to, the total reproduction time of the referred songs, text information, and information concerning still pictures referred to by the playlist.

0131

Each item will be described.

0132

PLI_ID contains an ID used for identifying PLI uniquely.

0133

PLI_TK_Ns contains the number of tracks that are referred to by the PLI. The maximum number is 99.

0134

PLI_PB_TM shows the total reproduction time of the all songs that are referred to by the playlist, in units of milliseconds.

0135

PLI_APP_ATR contains an application attribute for playlist.

0136

PLI_FCA is reserved for a future extension of the playlist.

0137

Now, the text information of the playlist will be described. As is the case with the default playlist, the playlist can have unique text information. The text information may be divided into two types according to Character set code.

0138

PLI_PLTI1_ATR contains text attribute information of playlist. As shown in FIG. 35, the bits from b7 to b0 contain Character set code. "00h" represents ISO646 (ASCII), "01h" JISX0201 (ASCII + Kana (a Japanese set of Characters)), and "02h" ISO 8859-1. The bits from b15 to b8 are reserved for a future extension.

0139

PLI_PLTI2_ATR contains text attribute information of the playlist. As shown in FIG. 35, the bits from b7 to b0 contain Character set code. "81h" represents Music Shift JIS Kanji. The bits from b15 to b8 are reserved for a future extension.

0140

PLI_PLTI contains the text information of the playlist. When there is no text, the text information is filled with "0".

When there is a text and the space prepared for the text information is not used entirely, the remaining space is filled with "0".

0141

When there are two texts having different character codes, the text information of the first text corresponding to DPLI_PLTI1_ATR is first written, the end code "0x00" is written, then the text information of the second text corresponding to DPLI_PLTI2_ATR is written. In this case also, when the space prepared for the text information is not used entirely, the remaining space is filled with "0".

0142

PLI_IOB_SRP contains the still picture (IOB) numbers of still pictures referred to by the default playlist and contains their attributes. 20 numbers are written there. As shown in FIG. 36, the bits from b25 to b16 indicate an IOB number. That is to say, the bits indicate one of 1 to 999 corresponding to an IOB file. When no still picture is referred to, "0" is written. The bits from b15 to b14 indicate a display timing mode. "00b" represents the slide show mode, "01b" the browsable mode. It should be noted here that in the slide show mode, the video synchronizes with the audio and the skip reproduction of only video is not possible. In the browsable mode, the video does not synchronize with the audio and the skip reproduction of only video is possible. The bits from b13 to b12 indicate the display order mode. "00b"

represents a sequential display, "01b" a random display, and "10b" a shuffle display. The bits from b11 to b8 indicate the image size. "0000b" represents 160*120, "0001b" 320*240, "0010b" 640*480, "0011b" 800*600, "0100b" 1024*768, and "0101b" 1280*1024. The bits from b7 to b4 indicate the number of colors in still pictures. "0000b" represents 24 bits, "0001b" 16 bits, and "0010b" 8 bits. The bits from b3 to b0 indicate the image coding mode. "0000b" represents JPEG (Joint Photograph Expert Group). The bits from b31 to b26 are reserved for a future extension.

0143

Up to now, the default playlist general information (DPLGI) has been described.

0144

Now, Playlist Track Search Pointer (PL_TK_SRP) will be described.

0145

Playlist Track Search Pointer (PL_TK_SRP) contains TKI reference information. Also, the description order of Playlist Track Search Pointer (PL_TK_SRP) indicates a reproduction order. During the reproduction process, a TKI to be reproduced is specified in accordance with the reference information. The number of prepared PL_TK_SRPs are 99. If there is no TKI to be referred to, PL_TK_SRPs are filled with "0".

0146

As shown in FIG. 40, the bits from b9 to b0 of PL_TKIN

indicate a TKI number. The number ranges from 1 to 999. This enables the reference target TKI to be specified. The bits from b15 to b10 are reserved for a future extension.

0147

Up to now, Default Playlist Track Search Pointer (DPL_TK_SRP), Default Playlist Information (DPLI), and Playlist Manager (PLMG) have been described.

0148

Now, Track Manager (TKMG) will be described.

0149

Track Manager contains information regarding tracks stored in the SD_AUDIO directory. As shown in FIG. 28, Track Manager is composed of a plurality of pieces of Track Information (TKI). The number of TKIs is 999 at the maximum. The following is a description of TKI.

0150

TKI is information used for managing the tracks. As shown in FIG. 41, TKI first contains Track General Information (TKGI), then Track Text Information Data Area (TKTXTI_DA), then Track Time Search Table (TKTMSRT).

0151

TKI has a fixed length (1536 B). TKGI and TKTXTI_DA have a fixed length of 512 bytes in total. TKTMSRT has a fixed length of 1024 bytes. TKI is information used for managing AOB_Block and AOB files.

0152

TKI is used in three ways as follows.

(1) One TKI Contains All Information for One Track

(2) A Plurality of TKIs Contain One Piece of Track Information (Part 1)

When one piece of Time Search information cannot be stored in the Track Time Search Map area in one TKI since one Track has a long reproduction period, the TKI continuation flag is turned On, then the Time Search information, as a continuation, is stored in the Track Time Search Map area in the next TKI. In this case, the same information is stored, except for the TKI continuation flag and Time Search information. Furthermore, the AOB file is divided.

(3) A Plurality of TKIs Contain One Piece of Track Information (Part 2)

When a plurality of Tracks are combined into one Track, the Track has a plurality of files storing audio information. In this case, reproduction of a song is achieved by continuously reproducing a plurality of AOB files that are referred to by the plurality of combined TKIs.

0153

Now, Track General Information (TKGI) will be described. As shown in FIG. 42, TKGI contains a song reproduction time, attribute information of AOB or IOB to be referred to, reference target infomation of AOB or IOB, and time search table reference

information. The following is a description of each item.

0154

TKI_ID contains an ID used for identifying a TKI uniquely.

0155

TKI_UI contains information indicating whether the TKI is used. As shown in FIG. 43, the bits from b1 to b0 indicate whether the TKI is used, i.e., whether the TKI is valid. "00b" indicates that the TKI is not valid. "01b" indicates that the TKI is valid.

0156

TKIN a TKI number that is one of 1 to 999. Note that the TKIN should not be the same as that of any other TKIs.

0157

TKI_SZ indicates the TKI size in units of bytes.

0158

TKI_LNK_PTR indicates a TKI number of a TKI to which the present TKI connects, when the track is composed of a plurality of TKIs.

0159

TKI_BLK_ATR indicates whether the TKI is used. As shown in FIG. 44, the Block Attribute composed of bits b2 to b0 indicates whether the TKI of a reference target is used or not.

0160

"000b" represents the case where the TKI is used, and one song is included in one TKI. "001b" represents the case where the TKI is used, one song is composed of a plurality of TKIs, and the TKI is the first one of the plurality of TKIs. "010b" represents the case where the TKI is used, one song is composed of a plurality of TKIs, and the TKI is a middle one of the plurality of TKIs. "011b" represents the case where the TKI is used, one song is composed of a plurality of TKIs, and the TKI is the last one of the plurality of TKIs. "100b" represents the case where the TKI is not used and space for TKI has been allocated, that is to say, it is a deleted TKI. "101b" represents the case where the TKI is not used and space for TKI has not been allocated, that is to say, it is TKI in the initial state. The bits from b15 to b3 are reserved for a future extension.

0161

TKI_PB_TM shows the reproduction time of the songs that are referred to by the TKI, in units of milliseconds.

0162

TKI_AOB_ATR contains TKI audio attribute. As shown in FIG. 45, the bits b13 to b11 indicate the coding mode. "000b" indicates that the encoding conforms to MPEG-2 AAC (with ADTS header). The bits b10 to b8 indicate the bit rate. "000b" indicates 64 kpps, "001b" 32 kpps, and "010b" 16 kpps. The bits b7 to b4 indicates the sampling frequency. "0000b" indicates 48 kHz, "0001b" 44.1 kHz, and "0010b" 32 kHz. The bits b3 to b1 indicate the number of channels.

"000b" indicates 1ch(mono), and "001b" 2ch(stereo). The bits b31 to b14 and bit b0 are reserved for a future extension.

0163

TKI_TI1_ATR contains TKI text attribute information. As shown in FIG. 35, the bits b7 to b0 indicate Character set code. "00h" represents ISO646 (ASCII), "01h" JISX0201 (ASCII + Kana (a Japanese set of Characters)), and "02h" ISO 8859-1. The bits from b15 to b8 are reserved for a future extension.

0164

TKI_TK2_ATR contains text attribute information of the TKI. As shown in FIG. 35, the bits from b7 to b0 contain Character set code. "81h" represents Music Shift JIS Kanji. The bits from b15 to b8 are reserved for a future extension.

0165

TKI_TMSRT_SA indicates the starting position of the time search table by a relative address from the TKI start, in units of bytes.

0166

ISRC shows the ISRC for TKGI in the format shown in FIG. 46. For detailed information of ISRC, refer to ISO3901: 1986 "Documentation-International Standard Recording Code (ISRC)".

0167

TKI_FCA is reserved for a future extension.

0168

The block information table (BIT) is a table used for

managing AOB_BLOCK.

0169

BIT is composed as shown in FIG. 48. Each component will be described.

0170

DATA_OFFSET shows the starting address of each AOB_BLOCK in units of bytes.

0171

SZ_DATA shows the starting address of each AOB_BLOCK in units of bytes.

0172

TMSRTE_Ns shows the total number of TMSRT_entry included in each AOB_BLOCK.

0173

TMSRT_OFFSET shows an offset for the starting address of AOB_BLOCK.

0174

Fns_1st_TMSRTE indicates the number of AOB_FRAMEs included in ADR_ST through the first TMSRT_entry, when one or more TST_entry's are included in the AOB_BLOCK. When there is no TMSRT_entry, FRAME_OFFSET is 0.

0175

Fns_Last_TMSRTE indicates the number of AOB_FRAMEs included in the last AOB_ELEMENT of the AOB_BLOCK.

0176

Fns_Middle_TMSRTE indicates the number of AOB_FRAMEs excluding those in the first and the last AOB_ELEMENTs. As shown in FIG. 49, the bits from b11 to b0 indicate the number of AOB_FRAMEs constituting the AOB_ELEMENTs. The bits from b15 to b12 are reserved for a future extension. Note that this value depends on the AOB sampling frequency value, as shown in FIG. 47.

0177

PLI_IOB_SRP contains IOB numbers and IOB attribute information referred to by TKI. 20 numbers are written there. As shown in FIG. 36, the bits from b25 to b16 indicate an IOB number. That is to say, the bits indicate one of 1 to 999 corresponding to an IOB file. When no still picture is referred to, "0" is written. The bits from b15 to b14 indicate a display timing mode. "00b" represents the slide show mode, "01b" the browsable mode. It should be noted here that in the slide show mode, the video synchronizes with the audio and the skip reproduction of only video is not possible. In the browsable mode, the video does not synchronize with the audio and the skip reproduction of only video is possible. The bits from b13 to b12 indicate the display order mode. "00b" represents a sequential display, "01b" a random display, and "10b" a shuffle display. The bits from b11 to b8 indicate the image size. "0000b" represents 160*120, "0001b" 320*240, "0010b" 640*480, "0011b" 800*600, "0100b" 1024*768, and "0101b" 1280*1024. The bits from b7 to b4 indicate the number of colors in still pictures. "0000b" represents 24 bits, "0001b" 16 bits, and "0010b" 8 bits. The

bits from b3 to b0 indicate the image coding mode. "0000b" represents JPEG (Joint Photograph Expert Group). The bits from b31 to b26 are reserved for a future extension.

0178

Up to now, Track General Information (TKGI) has been described.

0179

Now, Track Text Information Data Area (TKTXI_DA) will be described.

0180

Track Text Information Data Area (TKTXI_DA) contains TKI text information. Even if there is no text data, space is allocated to this information.

0181

In TKTI_DA, each piece of text data follows a tag for each item, as shown in FIG. 50. Each tag is followed by text data, then the end code.

0182

As shown in FIG. 50, the tag "01h" indicates a title name, the tag "02h" an artist name, the tag "03h" an album name, the tag "04h" a songwriter, the tag "05h" a composer, the tag "06h" an arranger, the tag "07h" a producer, the tag "08h" a record company, the tag "09h" an artist's message, the tag "0Ah" a user's comments, the tag "0Bh" a provider's comments, the tag "0Ch" year, month, day, the tag "0Dh" a genre, the tag "0Eh" an URL (Uniform Resource

Locator), the tag "0Fh" a free item (item that can be set by the user) 1, the tag "10h" a free item 2, the tag "11h" a free item 3, the tag "12h" a free item 4, the tag "13h" a free item 5, and the tag "14h" a free item 6.

0183

The end code will be described. "0x00" indiates ISO646, JISX0201, ISO8859-1; and "0x0000" indiates Music Shift JIS Kanji.

0184

Each text for the above 20 items has a variable length. The total size of TKTXTI_DA is 256.

0185

Up to now, TKTXTI_DA has been described.

0186

Now, the time search table (TMSRT) will be described.

The time search table manages the address information that is provided approximately every 2 seconds. The time search table, having a fixed length of 1024 bytes, is used for calculating and displaying time during the fastforward or rewinding operation. When the size of the time search map for one song exceeds 1024 bytes, TKI and AOB file are newly created, and the time search map for the created one is used.

0187

As shown in FIG. 51, one TMSRT is provided for one AOB_BLOCK. Each TMSRT is composed of a time search table header

and a plurality of pieces of TMSRT_entry.

0188

The time search table header (TMSRT_H) will be described.

0189

The time search table header (TMSRT_H) is placed at the start of a TMSRT, and contains information relating to the whole TMSRT. FIG. 52 shows a detailed data structure of the time search table header (TMSRT_H).

0190

TMSRT_ID contains an ID that is used to uniquely identify TMSRT.

0191

The Total TMSRT_entry Number contains the total number of pieces of TMSRT_entry in the TMSRT.

0192

TMSRT_ENT contains the starting address of AOB_ELEMENT, as shown in FIG. 53.

0193

Up to now, the time search map (TMSRMap) has been described.

0194

This completes the description of Track Manager (TKMG).

0195

Now, IOB Manager (IOBMG) will be described.

0196

As shown in FIG. 29, IOBMG contains information for managing IOB. IOBMG first includes the IOB management information (IOBMGI), then IOB Count Information (IOBCI). Each component will be described.

0197

As shown in FIG. 54, the IOB management information (IOBMGI) contains the identification information of IOBMGI and the number of IOBs.

0198

IOBMGI_ID contains an ID used for uniquely identifying IOBMGI.

0199

IOB_Ns contains the number of IOBs.

0200

Up to now, the IOB management information (IOBMGI) has been described.

0201

Now, IOB Count Information (IOBCI) will be described. As shown in FIG. 55, the IOB Count Information (IOBCI) is composed of IOB_RCNs, and indicate whether each IOB is referred to by the default playlist, playlist, or track. When an IOB is referred to, the number of references is written there.

0202

As shown in FIG. 56, the bits from b9 to b0 of IOB_RCN

indicate one of 1 to 999 as the number of references by the default playlist, playlist, or track, for each IOB. When the IOB is not referred to, 0 is written there. The bits from b15 to b10 are reserved for a future extension.

0203

Up to now, IOBManager (IOBMG) has been described.

This completes the description of the navigation data.

0204

Now, the editing operation using the default playlist will be described.

0205

Default Playlist Information (DPLI) in Playlist Manager includes Track Information (TKI) and information for managing the file. DPL_TK_ATR of DPLI indicates the state of TKI, and DPL_TKN indicates numbers that are allocated to TKI and the file.

0206

As shown in FIG. 5, songs A, B, C, and E are each stored in one TKI, though a long song such as song D is stored in a plurality of TKIs.

0207

This is because since the time search table (TMSRT) in the TKI has a fixed length of 1024 bytes, when one TMSRT is not enough to store data for a song, TMSRTs of a plurality of TKIs should be used.

0208

When a song is stored in a plurality of TKIs, the search tables DPL_TK_SRP corresponding to the plurality of TKIs are continuously written, and DPL_TK_ATR contains information indicating the state of each TKI.

0209

FIG. 6 shows the attributes of DPL_TK_ATR.

0210

In an example shown in FIG. 5, song D is stored in TKI_4 through TKI_7 in division. Each TKI links to the next TKI. DPL_TK_ATR indicates the head of song, midpoint of song, and end of song.

0211

FIG. 7 shows deletion of songs.

0212

When the user deletes a song from Playlist, a reference pointer to TKI of the Playlist is deleted. When in reality a song is deleted from SD-AUDIO, the song should be deleted from the default Playlist. This is performed in accordance with the following operational flow.

0213

1. Entries are deleted from DPL_TK_SRP of the song specified by the user. This deletion is done by setting DPL_TK_ATR to "unused", and moving it to the last of DPL.

0214

2. A TKI with a number indicated by DPL_TKN is set to "unused".

0215

3. An AOB file and an IOB file with a number indicated by DPL_TKN are deleted.

0216

4. When a song is managed by a plurality of TKIs and a plurality of files, the TKIs and files are deleted.

0217

FIG. 8 shows recording of songs.

0218

When the user records music in units of songs, information of new songs is added only to the default playlist. The operation in reality is executed in accordance with the following operational flow.

0219

1. When the user starts to record music, the DPL is searched for an unused entry of DPL_TK_SRP.

0220

2. An entry of TKI and an AOB file number are determined from the unused entry.

0221

3. Data of the music to be recorded is recorded in AOB files.

0222

4. The time search table of the data of the music to be recorded is written into the TKI time search table.

0223

5. When the TKI cannot store the time search table, another unused entry of DPL_TK_SRP is searched for, and a TKI and an AOB file are newly determined.

0224

6. Data except for TST is copied into the new TKI. A link is established between the formerly used TKI and the new TKI.

0225

7. The steps 3 to 6 are repeated. After the recording completes, the the head of song, midpoint of song, and end of song are written as flags into DPL_TK_ATR.

0226

FIG. 9 shows interchanging of songs.

0227

When the user interchanges songs in Default Playlist, the following operational flow is performed.

0228

1. Entries of DPL_TK_SRP for the songs specified by the user are interchanged.

0229

2. When a song is managed by a plurality of TKIs and files, the interchanging is performed so that the entries of DPL_TK_SRP for the song are continuous.

0230

As described above, even if songs in Default Playlist are

interchanged, it does not affect the order of songs in Playlist that refers to the song. As a result, when the user is to interchange songs in Playlist, the interchanging is achieved just by interchanging the reference pointers.

0231

FIG. 10 shows combining of songs.

0232

When the user combines two songs into one, Default Playlist and TKMG are processed in accordance with the following operational flow.

0233

1. The DPL_TK_SRP entries of the two songs specified by the user are disposed in succession.

0234

2. DPL_TK_ATR is rewritten to include the head of song, end of song or the like so that the two songs are stored as one song.

0235

The link in TKI is changed to the TKI number of the song generated by the combination.

0236

FIG. 11 shows dividing of songs.

0237

When a song is divided into two two, Default Playlist and TKMG are processed in accordance with the following operational flow.

0238

1. DPL is searched for an unused entry. An unused TKI is searched for.

0239

2. An unused DPL_TK_SRP entrry is moved to immediately after the song to be divide.

0240

3. TKI is obtained from the DPL_TK_SRP entrry to be divided. Data for the song except for the time search table is copied into the unused TKI.

0241

4. The time search table of the TKI to be divided and the AOB file are divided, and assigned to the unused TKI and file.

0242

5. The unused TPL_TK_ATR and TKI are changed to "in use".

0243

The song is divided and the meaning of TKI in TKMG changes through the above process. However, there is no need of adding TKIs of the newly generated songs to the reference pointers of Playlist referring to the TKI of the song to be divided

0244

As described above, by using two types of playlists: a default playlist with TKI management function; and a user playlist that can specify a reproduction order, it is possible to prevent song editing from affecting the user playlist. Also, it is also

possible to prevent song editing from affecting TKIs by making each TKI to have a fixed length.

0245

It should be noted here that the above embodiment only provides an example of a system that is considered to present the best effects. However, the present invention can be achieved in various forms. The following are such examples.

0246

In the above embodiment, a semiconductor memory (media card) is used as the recording medium. However, not limited to this, optical discs such as DVD-RAM or hard disks can also be used as the recording medium.

0247

In the above embodiment, AAC is used as music data. However, not limited to this, MP3 (MPEG 1 Audio Layer 3), Dolby-AC3, or DTS (Digital Theater System) can also be used as music data.

0248

[EFFECTS OF THE INVENTION]

As described above, by using two types of information: a type of information for managing reproduction time of encoded data; and another type of information for managing files of encoded data, it is possible to minimize the changes caused by division or combination of the files of encoded data. This provides a great practical effect in that the present invention can be contained

with ease in apparatuses (e.g., a portable player) that have a small memory capacity.

[BRIEF DESCRIPTION OF THE DRAWINGS]

FIG. 1 shows the shape of the recording medium in an embodiment of the present invention.

FIG. 2 shows the construction of the areas in the recording medium in an embodiment of the present invention.

FIG. 3 shows the construction of the file system in the recording medium in an embodiment of the present invention.

FIG. 4 shows the construction of the directories and files in the recording medium in an embodiment of the present invention.

FIG. 5 shows relationships between DPL, TKI, and files in the recording medium in an embodiment of the present invention.

FIG. 6 shows the attributes of DPL_TK_ATR in the recording medium in an embodiment of the present invention.

FIG. 7 shows deletion of songs in the recording medium in an embodiment of the present invention.

FIG. 8 shows recording of songs in the recording medium in an embodiment of the present invention.

FIG. 9 shows interchanging of songs in the recording medium in an embodiment of the present invention.

FIG. 10 shows combining of songs in the recording medium in an embodiment of the present invention.

FIG. 11 shows dividing of songs in the recording medium in an embodiment of the present invention.

FIG. 12 shows the structure of presentation data in an embodiment of the present invention.

FIG. 13 shows relationships between the number of AOB_FRAMEs constituting AOB_ELEMENT and sampling frequency in an embodiment of the present invention.

FIG. 14 shows the structure of AOB in an embodiment of the present invention.

FIG. 15 shows restrictions for the MPEG2-AAC LC profile in an embodiment of the present invention.

FIG. 16 shows the structure of IOB in an embodiment of the present invention.

FIG. 17 shows the contents of IOB_ATR in an embodiment of the present invention.

FIG. 18 shows an example of TMSRT in an embodiment of the present invention.

FIG. 19 shows the entry relationships between AOB division and BIT in an embodiment of the present invention.

FIG. 20 shows the selection of a playlist in an embodiment of the present invention.

FIG. 21 shows a song reproduction order in an embodiment of the present invention.

FIG. 22 shows TKI reference information in an embodiment

of the present invention.

FIG. 23 shows an AOB division (FNs_Middle_TMSRTE=96) in an embodiment of the present invention.

FIG. 24 shows an example of changing TMSRT in correspondence with FIG. 13 in an embodiment of the present invention.

FIG. 25 shows an example of changing BIT in correspondence with FIG. 13 in an embodiment of the present invention.

FIG. 26 shows a system model shows an example of changing TMSRT in correspondence with FIG. 13 in an embodiment of the present invention.

FIG. 27 shows the structure of Playlist Manager (PLMG) in an embodiment of the present invention.

FIG. 28 shows the structure of Track Manager (TKMG) in an embodiment of the present invention.

FIG. 29 shows the structure of IOB Manager (IOBMG) in an embodiment of the present invention.

FIG. 30 shows the structure of Playlist Manager Information (PLMGI) in an embodiment of the present invention.

FIG. 31 shows a detailed data structure of VERN in an embodiment of the present invention.

FIG. 32 shows a detailed data structure of PLMG_AP_PL and PLMG_RSM_PL in an embodiment of the present invention.

FIG. 33 shows the structure of Default Playlist Information (DPLI) in an embodiment of the present invention.

FIG. 34 shows the structure of Default Playlist General Information (DPLGI) in an embodiment of the present invention.

FIG. 35 shows a detailed data structure of DPLI_PLTI1_ATR, DPLI_PLTI2_ATR, PLI_PLTI1_ATR, PLI_PLTI2_ATR, TKI_TI1_ATR, and TKI_TI2_ATR in an embodiment of the present invention.

FIG. 36 shows a detailed data structure of DPLI_IOB_SRP, PLI_IOB_SRP, and TKI_IOB_SRP in an embodiment of the present invention.

FIG. 37 shows a detailed data structure of DPL_TK_SRP in an embodiment of the present invention.

FIG. 38 shows the structure of Playlist Information (PLI) in an embodiment of the present invention.

FIG. 39 shows the structure of Playlist General Information (PLGI) in an embodiment of the present invention.

FIG. 40 shows a detailed data structure of PL_TK_SRP in an embodiment of the present invention.

FIG. 41 shows the structure of Track Information (TKI) in an embodiment of the present invention.

FIG. 42 shows the structure of Track General Information (TKGI) in an embodiment of the present invention.

FIG. 43 shows a detailed data structure of TKI_UI in an embodiment of the present invention.

FIG. 44 shows a detailed data structure of TKI_BLK_ATR in an embodiment of the present invention.

FIG. 45 shows a detailed data structure of TKI_AOB_ATR in

an embodiment of the present invention.

FIG. 46 shows a detailed data structure of ISRC in an embodiment of the present invention.

FIG. 47 shows relationships between sampling frequency and FNs_Middle_TMRTE in an embodiment of the present invention.

FIG. 48 shows the structure of Block Information Table (BIT) in an embodiment of the present invention.

FIG. 49 shows a detailed data structure of FNs_Middle_TMRTE in an embodiment of the present invention.

FIG. 50 shows relationships between the tag names and values for TKTXTI_DA in an embodiment of the present invention.

FIG. 51 shows the structure of TMSRT in an embodiment of the present invention.

FIG. 52 shows the structure of TMSRT_H in an embodiment of the present invention.

FIG. 53 shows the structure of TMSRT_entry in an embodiment of the present invention.

FIG. 54 shows the structure of IOBMGI in an embodiment of the present invention.

FIG. 55 shows the structure of IOBCI in an embodiment of the present invention.

FIG. 56 shows a detailed data structure of IOB_RCN in an embodiment of the present invention.

[DESCRIPTION OF CHARACTERS]

201 personal computer

202 media card

203 player

221 special area

222 authentication area

223 user area

[DOCUMENT] Abstract

[SUMMARY]

[AIM] To achieve a method for minimizing the changes caused by division of the files of encoded data by using two types of information: a type of information for managing reproduction time of encoded data; and another type of information for managing files of encoded data, in a semiconductor memory for recording music data, character data, etc.

[MEANS TO ACHIEVE THE AIM] The present data management method manages the information for managing reproduction time of encoded data, in units of 1KB-blocks. When a large piece of long-play music data that cannot be recorded in one block is divided so that it is also recorded in other files and managed. When a plurality of pieces of music data are combined, the music data files are not physically combined, but are managed in the state of separate fragments.

[SELECTED FIGURE] FIG. 5

FIG. 1

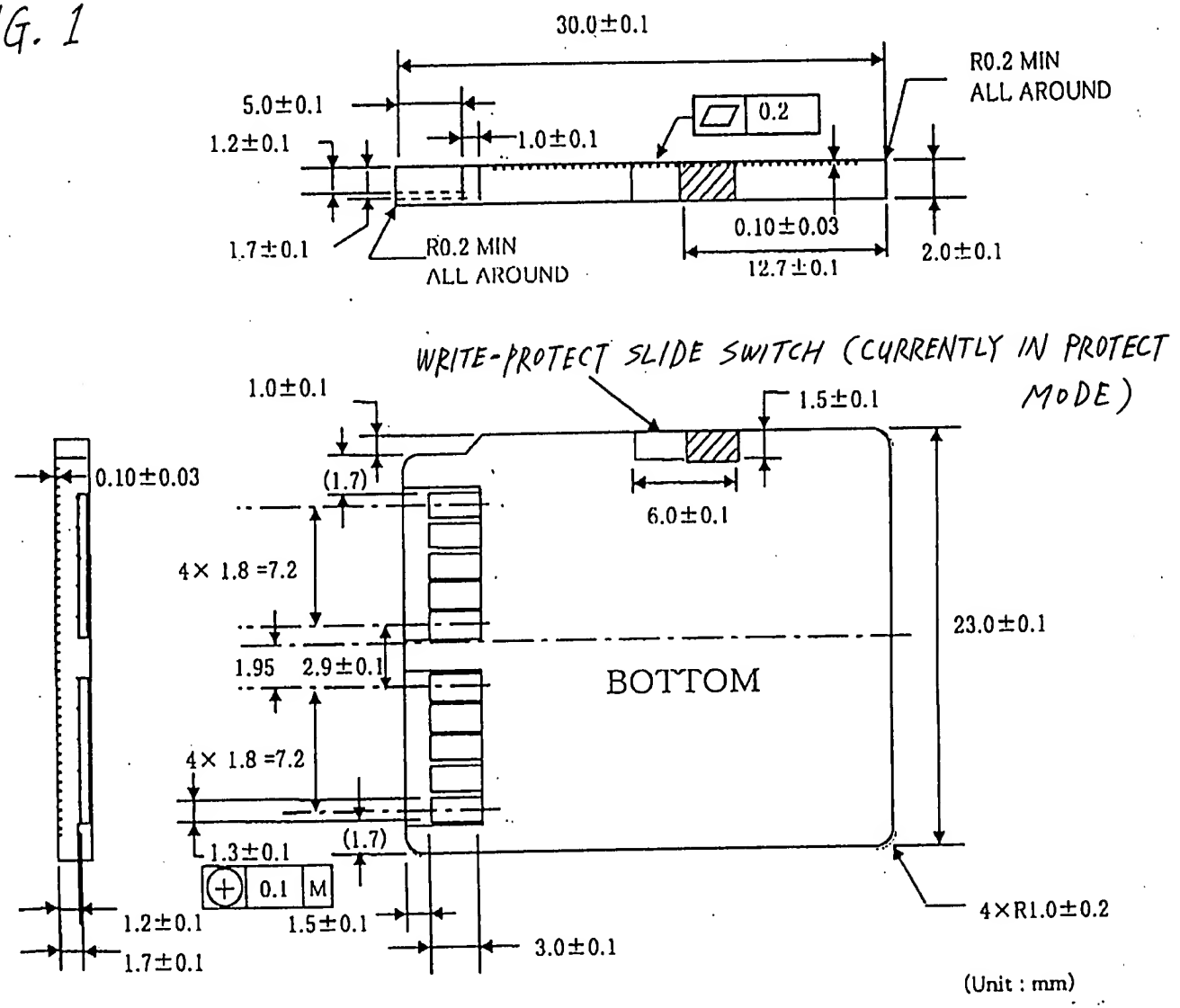


FIG. 2

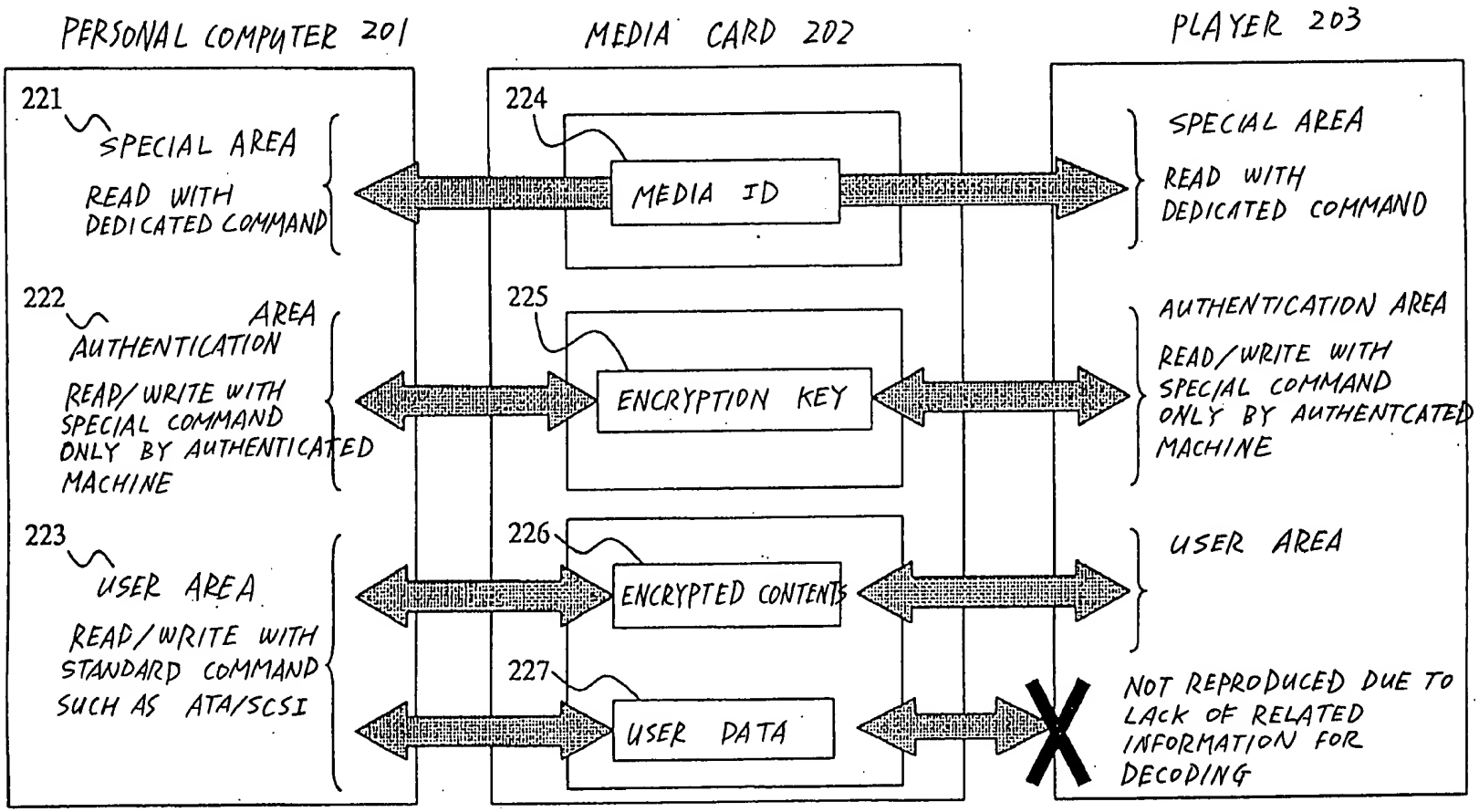


FIG. 3

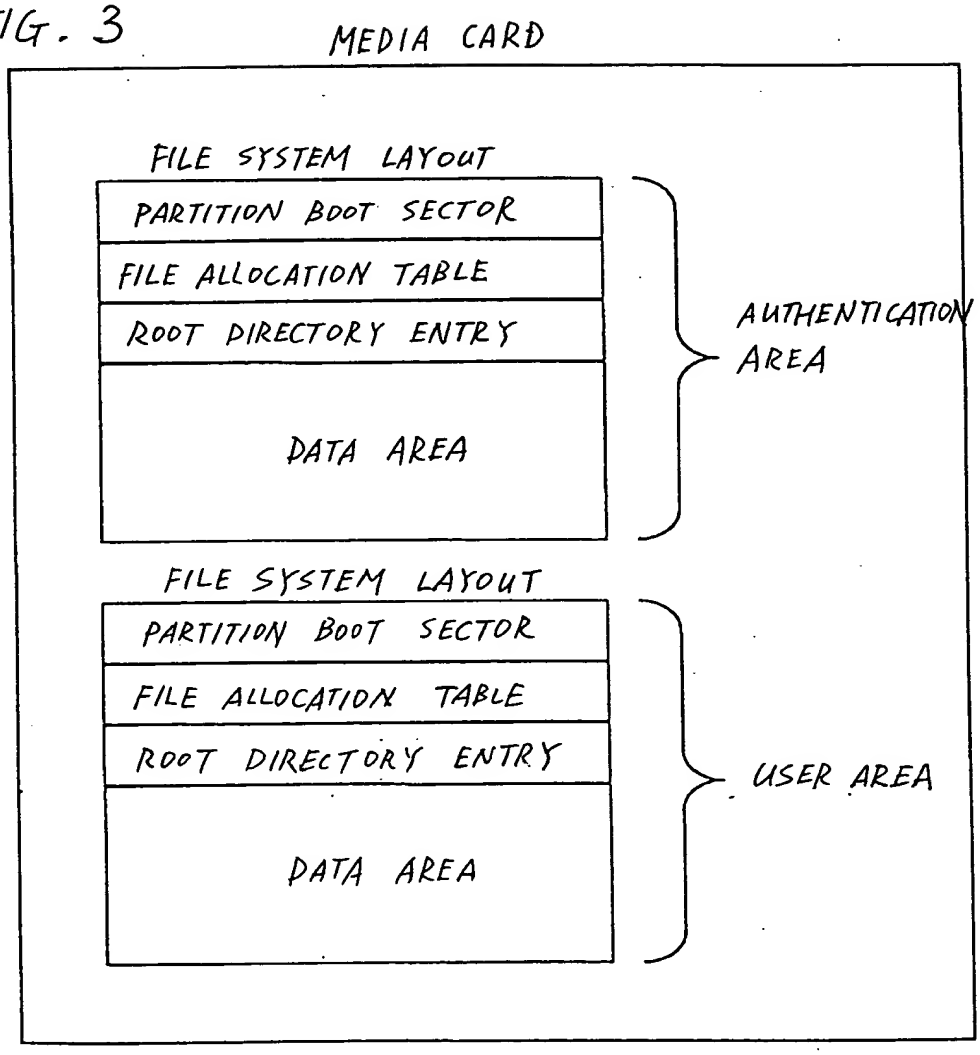


FIG. 4

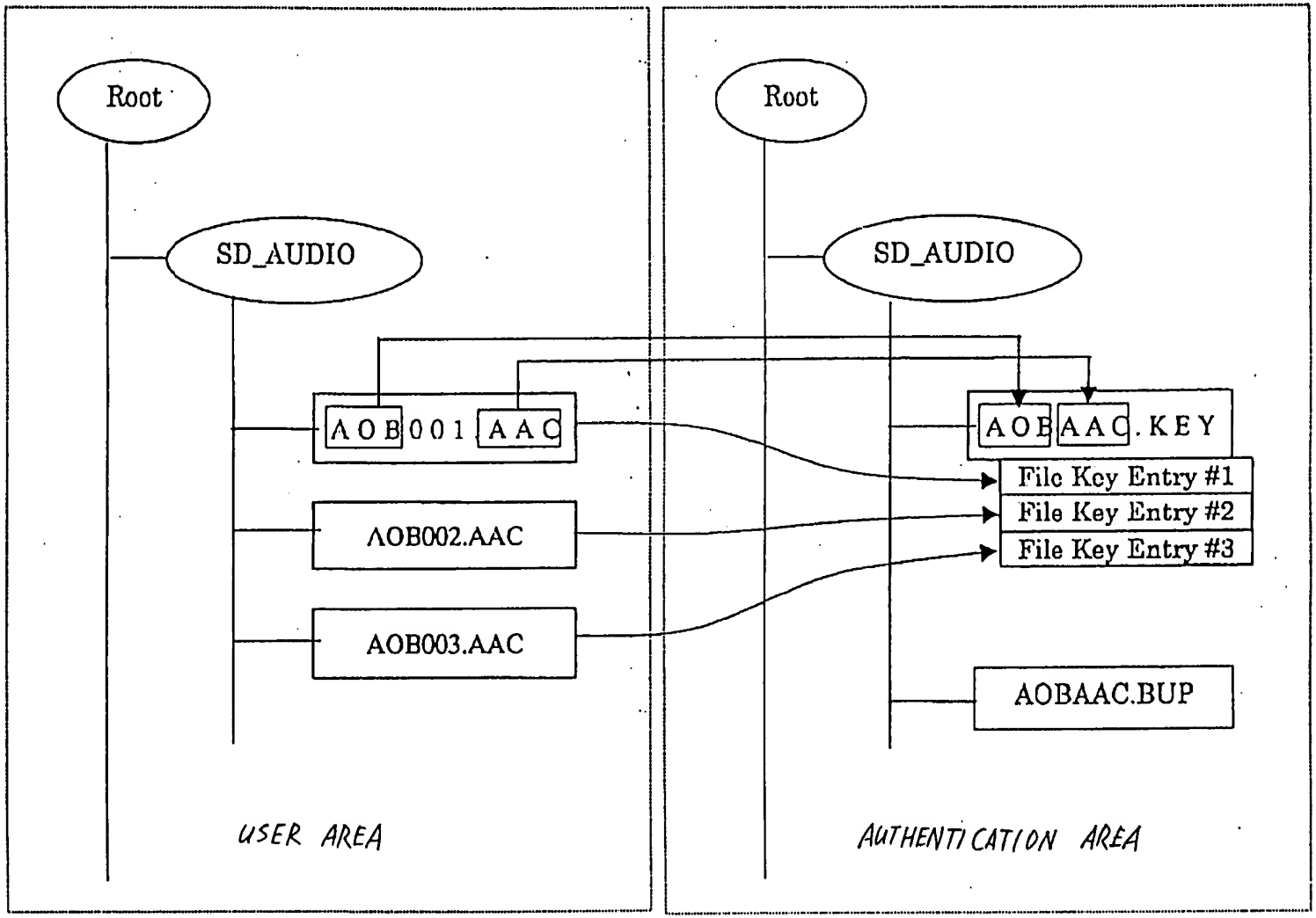


FIG. 5

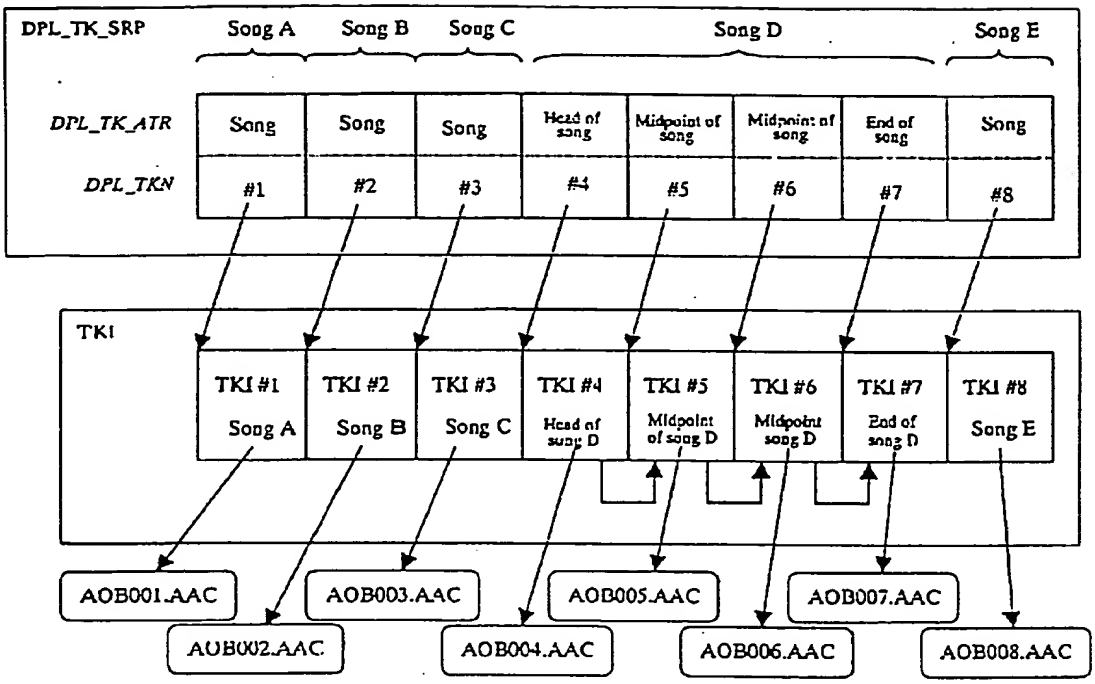


FIG. 6

ATTRIBUTES OF
DPL_TK_ATR

| ATTRIBUTE | INDICATION |
|---|---|
| SONG | ONE SONG IN ONE TKI |
| HEAD | ONE SONG IN PLURALITY OF TKIs. HEAD OF THE SONG. |
| MIDPOINT | ONE SONG IN PLURALITY OF TKIs. MIDPOINT OF THE SONG. |
| END | ONE SONG IN PLURALITY OF TKIs. END OF THE SONG. |
| UNUSED | TKI UNUSED. AREA FOR TKI ALLOCATED. ALLOCATED AT DELETION. |
| UNALLOC | TKI UNUSED. NO AREA FOR TKI. INITIALIZED. |

FIG. 7

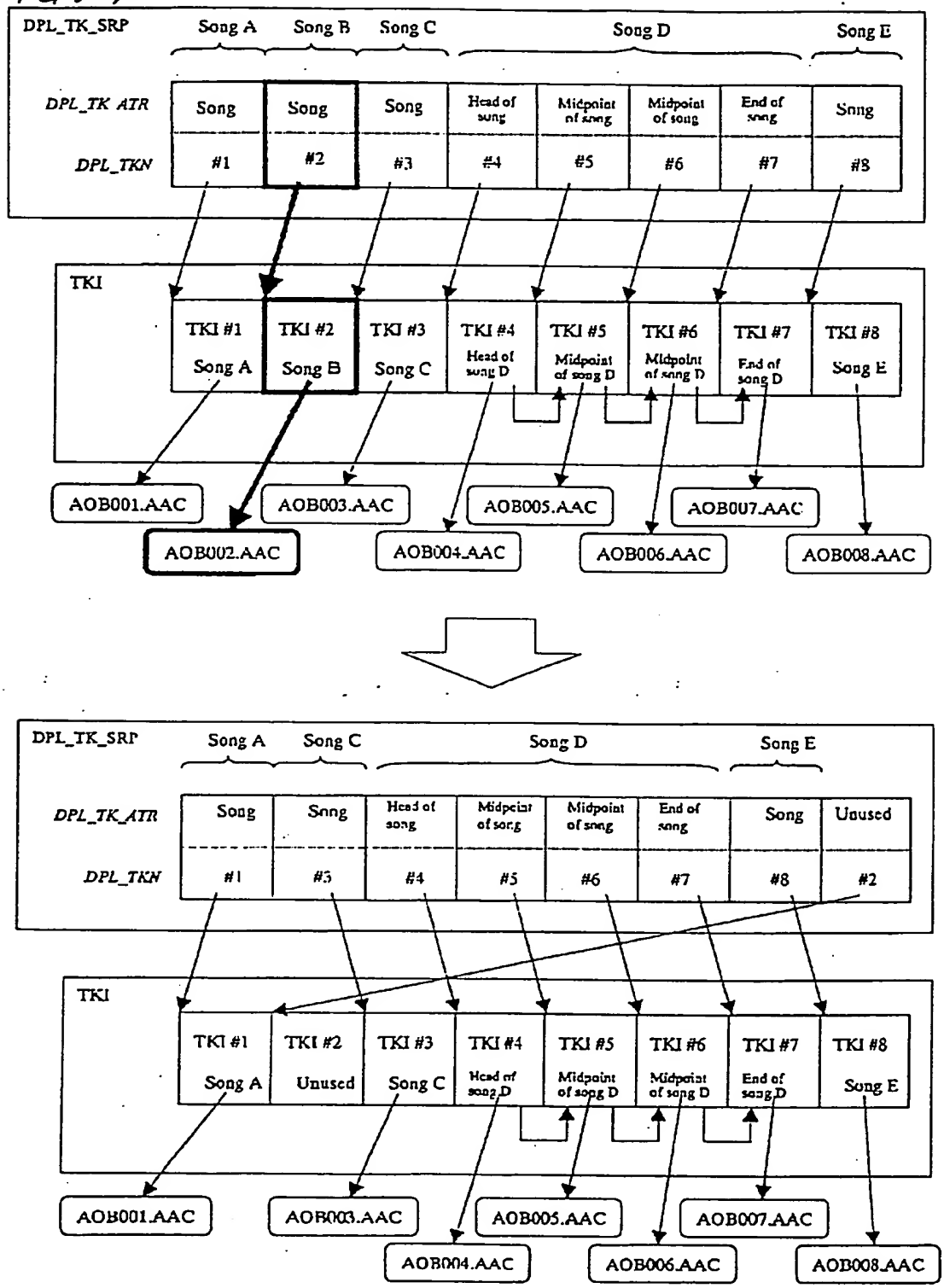


FIG. 8

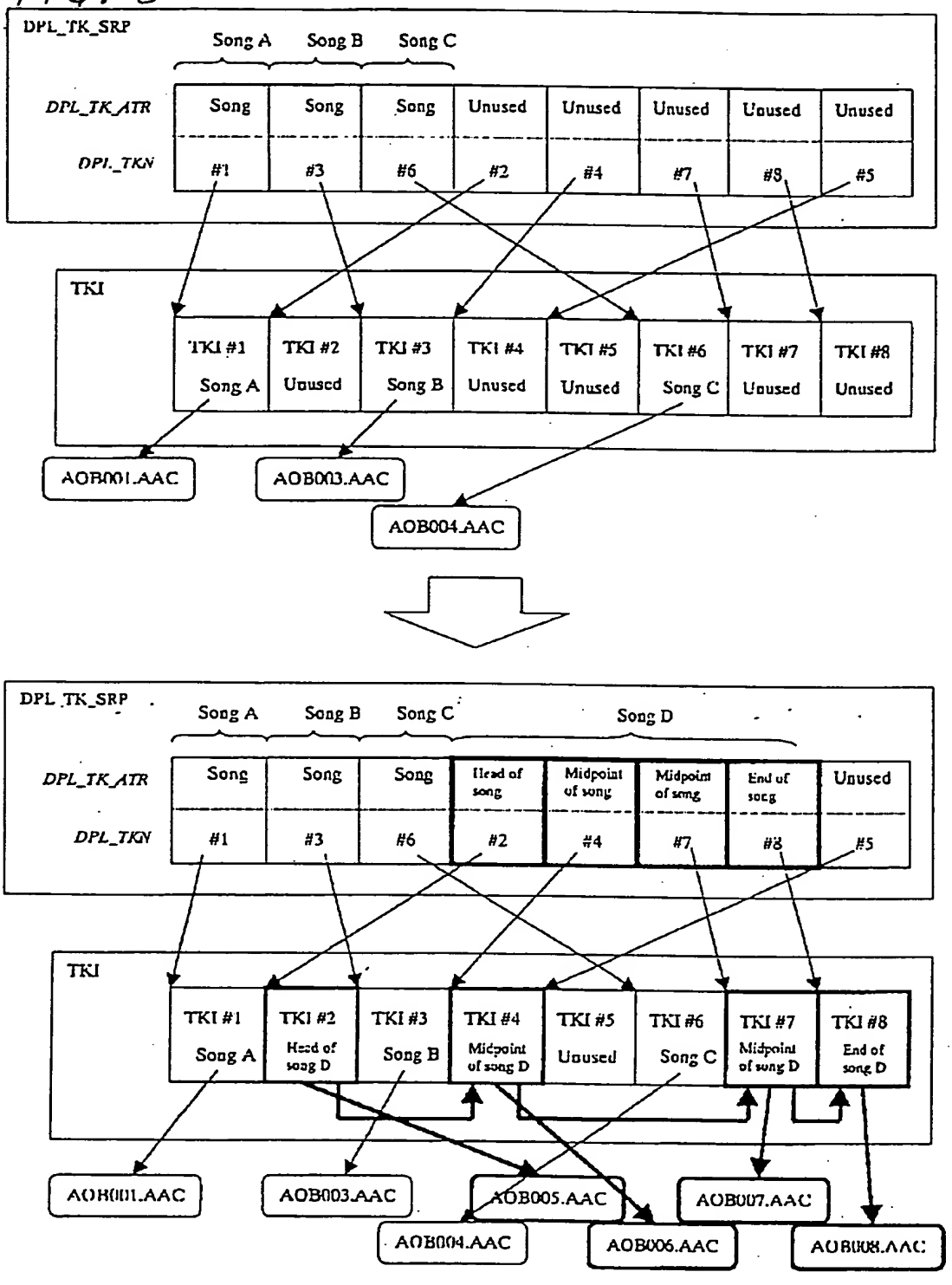


FIG. 9

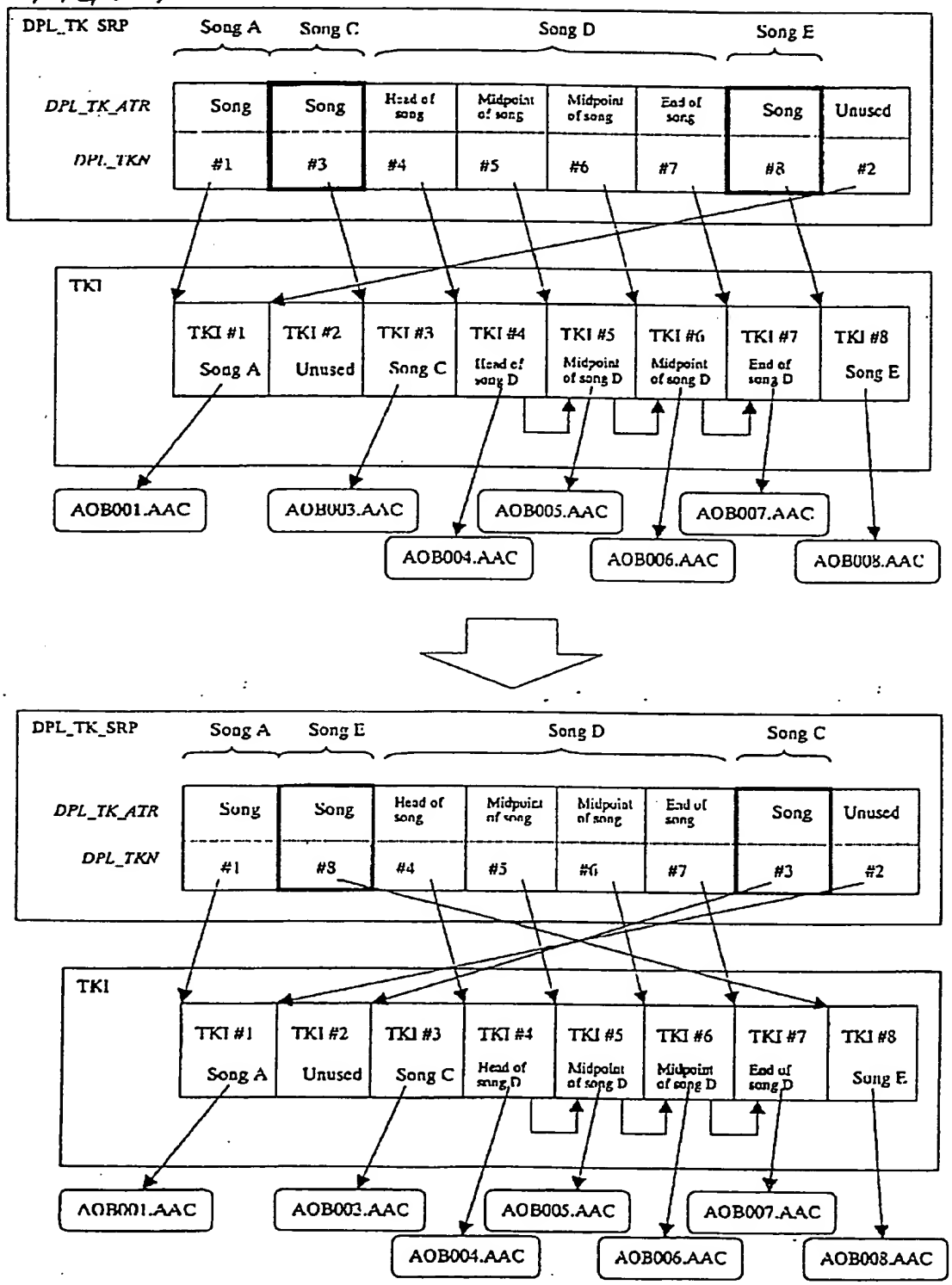


FIG. 10

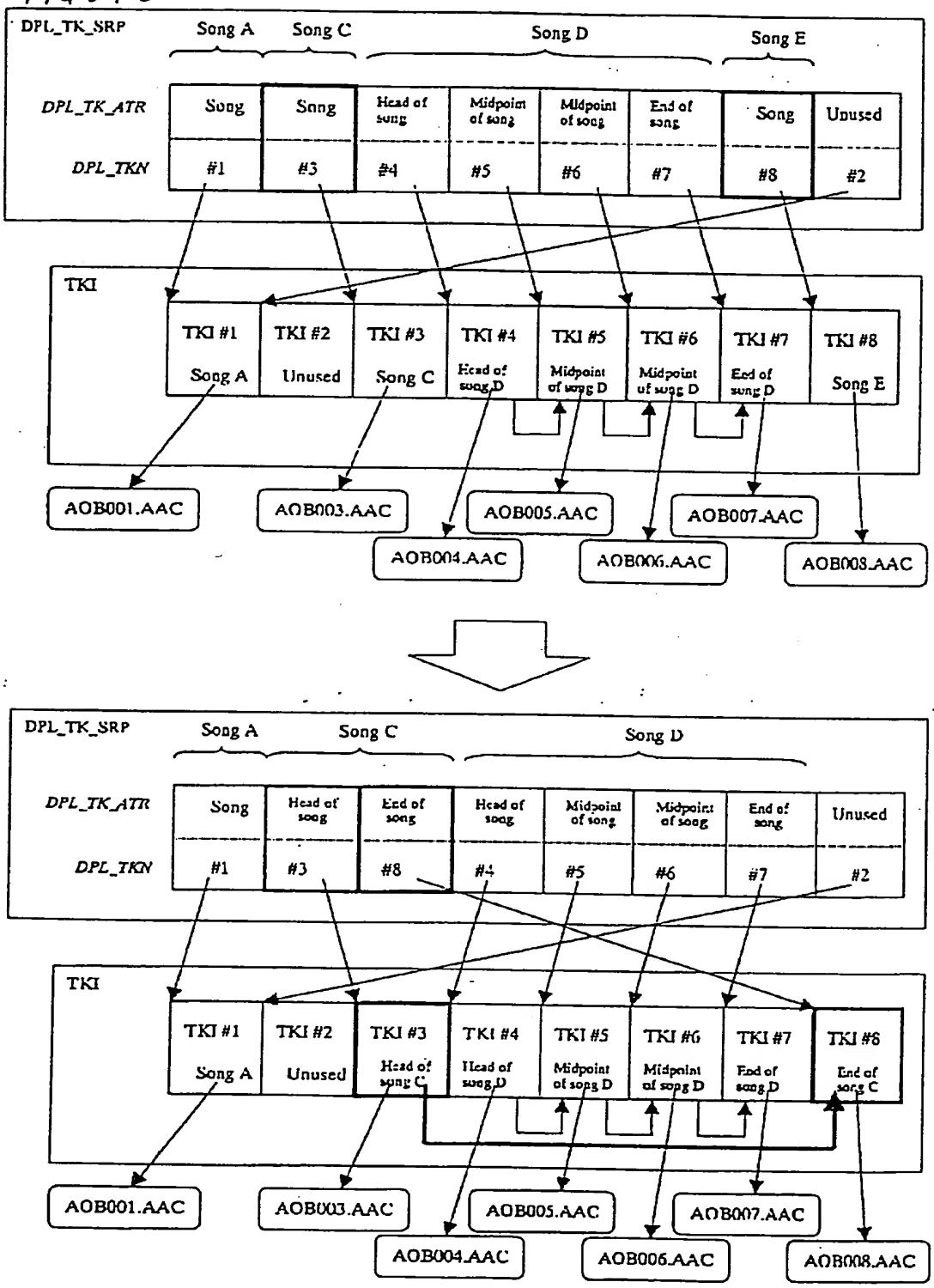


FIG. 11

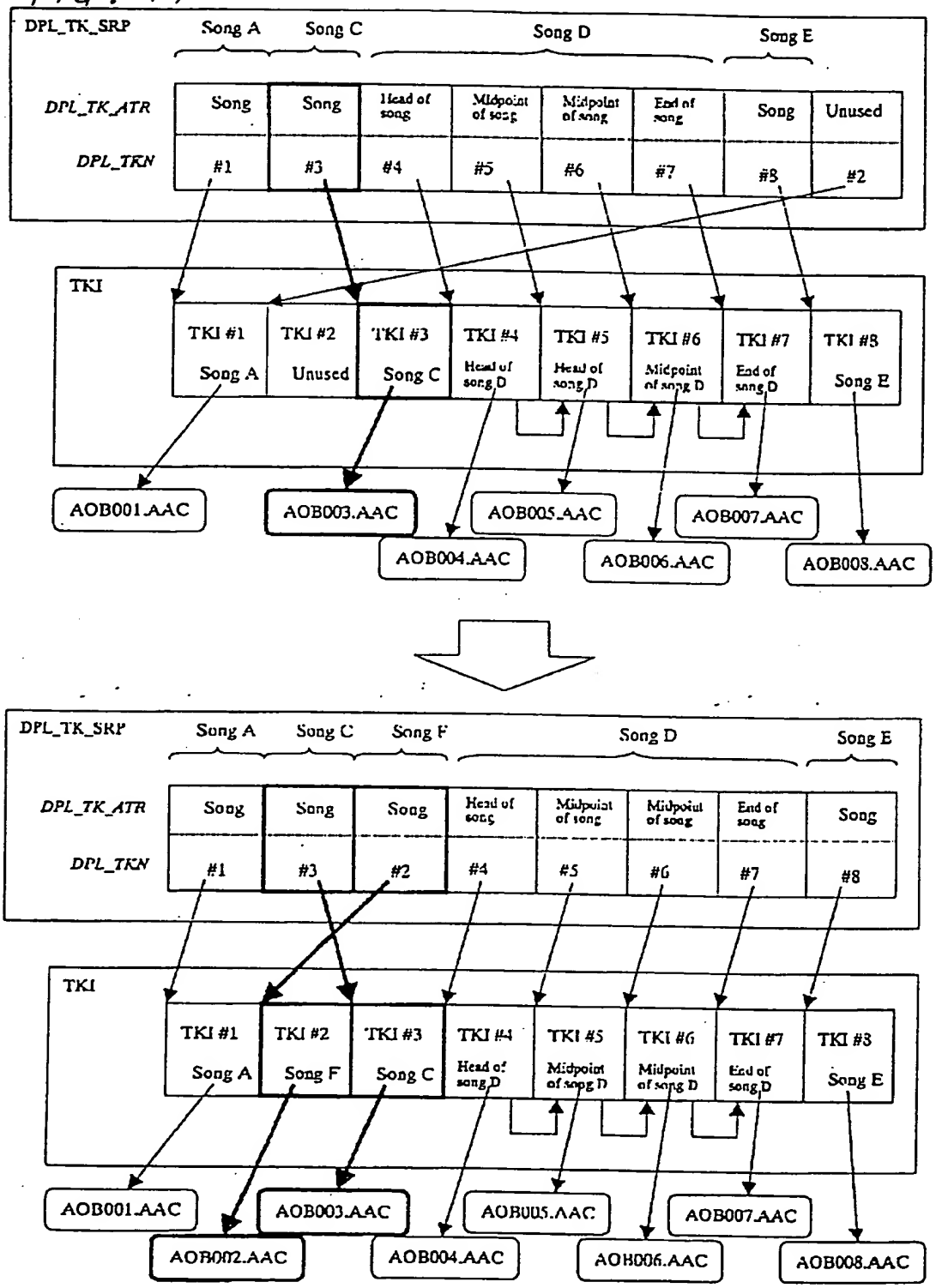


FIG. 12

| Audio Object (AOB) (mandatory) |
| --- |
| Image Object (IOB) (optional) |

FIG. 13

| SAMPLING FREQUENCY | THE NUMBER OF AOB_FRAMES |
| --- | --- |
| 48kHz | 96 |
| 44.1kHz | 88 |
| 32kHz | 64 |

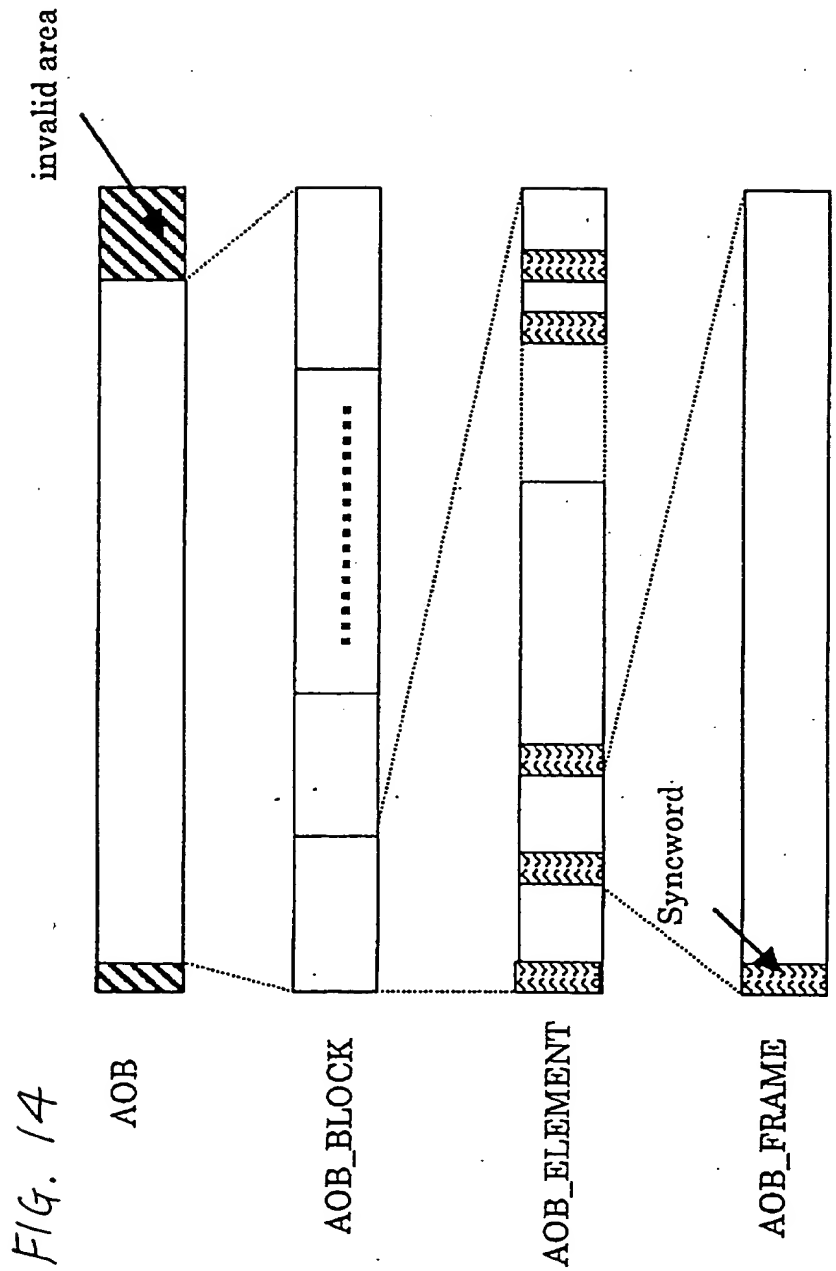



FIG. 14

FIG. 15

| Parameter | Value | Comments |
|---|---|---|
| profile | 01 | LC profile(mandatory) |
| sampling_frequency_index | 0011 | 48 kHz(mandatory) |
| | 0100 | 44.1 kHz(mandatory) |
| | 0101 | 32 kHz(mandatory) |
| | 0110 | 24 kHz |
| | 0111 | 22.05kHz |
| | 1000 | 16kHz |
| | 1001 | 12kHz |
| | 1010 | 11.025kHz |
| | 1011 | 8kHz |
| | others | optional |
| channel_configuration | 001 | single_channel_element(mandatory) |
| | 010 | channel_pair_element(mandatory) |
| | others | optional |
| number_of_raw_data_blocks_in_frame | 00 | 1 header / 1 raw_data_block(mandatory) |

FIG. 16

| RBP | Field Name | Contents | Numbers of bytes |
|---|---|---|---|
| 0 to 1 | IOB_ID | IOB FILE MAGIC NUMBER | 2 bytes |
| 2 | reserved | reserved | 1 bytes |
| 3 | IOB_ATR | INDIRECT REFERENCE FLAG | 1 bytes |
| 4 to 7 | IOB_SZ | IOB DATA LENGTH | 4 bytes |
| Total | | | 8 bytes |

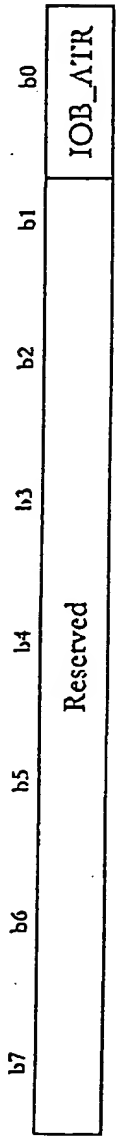
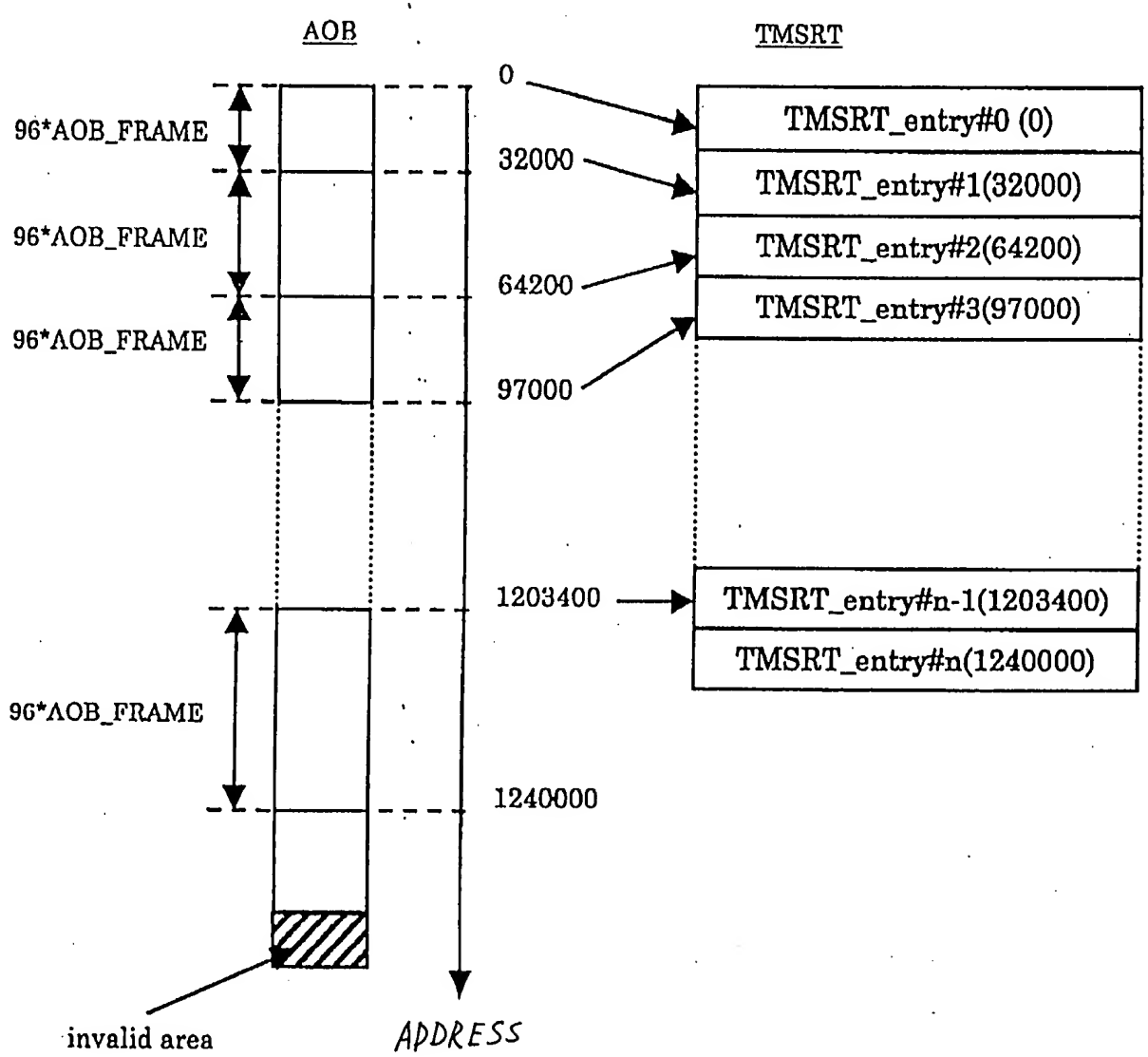



FIG. 17

FIG. 18



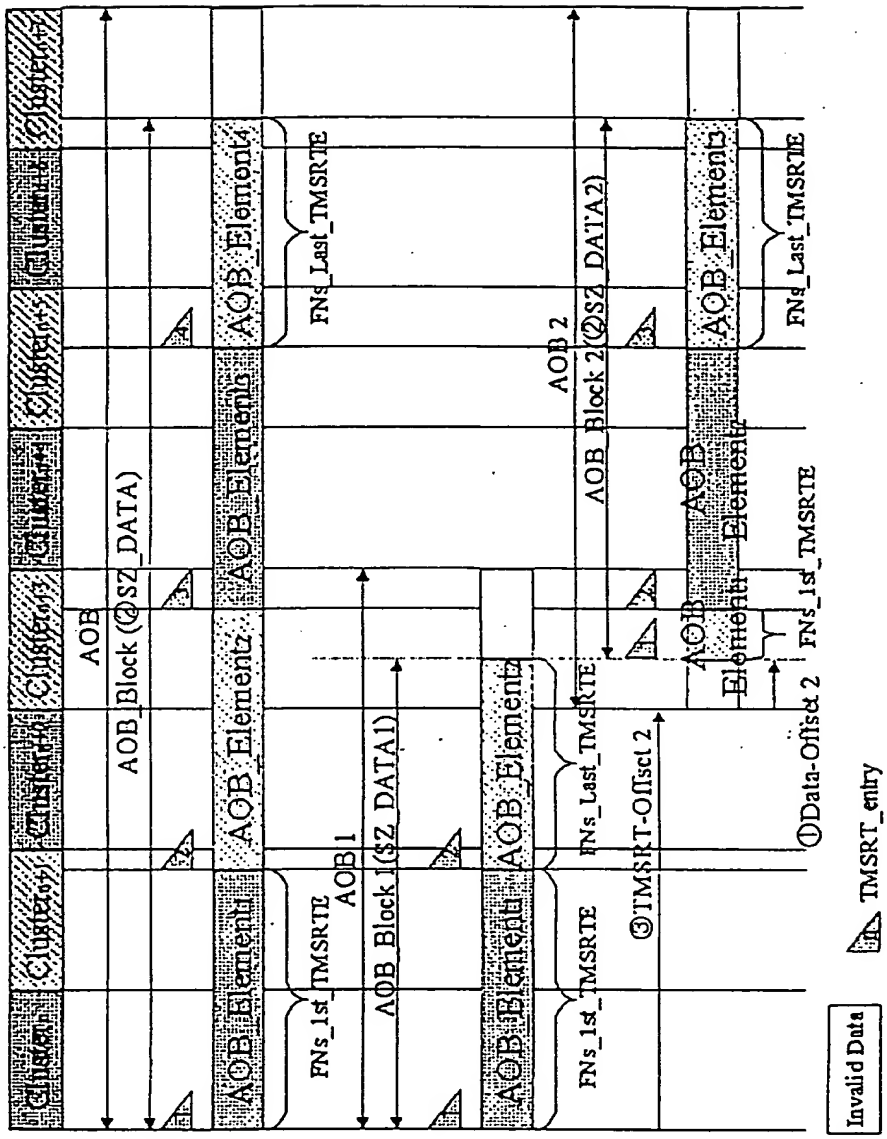



FIG. 19

FIG. 20

| PLAYLISTS REGISTERED WITH THE MEDIA | PLAYLIST NAME (EXAMPLE) |
|---|---|
| DEFAULT PLAYLIST (M) | *My Card* |
| PLAYLIST 1(O) | *Favorites* |
| PLAYLIST 2(O) | *'99 HITS* |
| PLAYLIST 3(O) | *Star Wars* |
| : | : |

(M:Mandatory; O:Optional)

FIG. 21

| PLAYLIST |
|---|
| SONG 1 |
| SONG 2 |
| SONG 3 |
| : |

FIG. 22

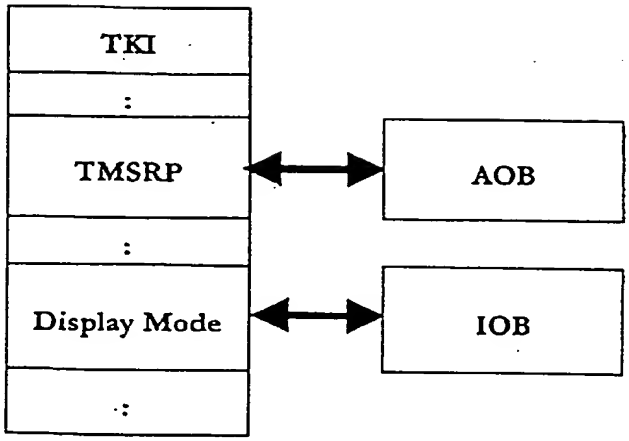


FIG. 23

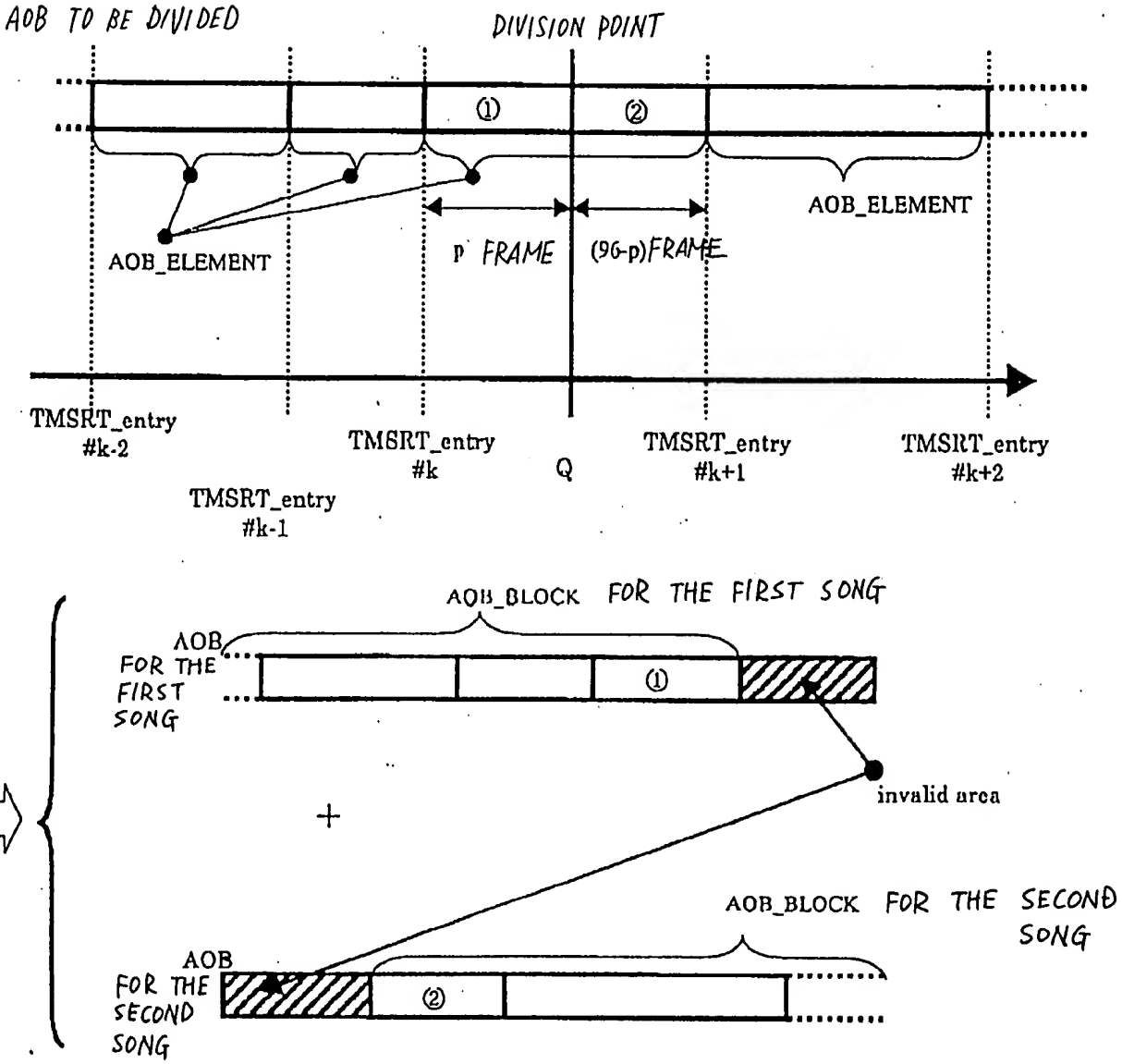


FIG. 24

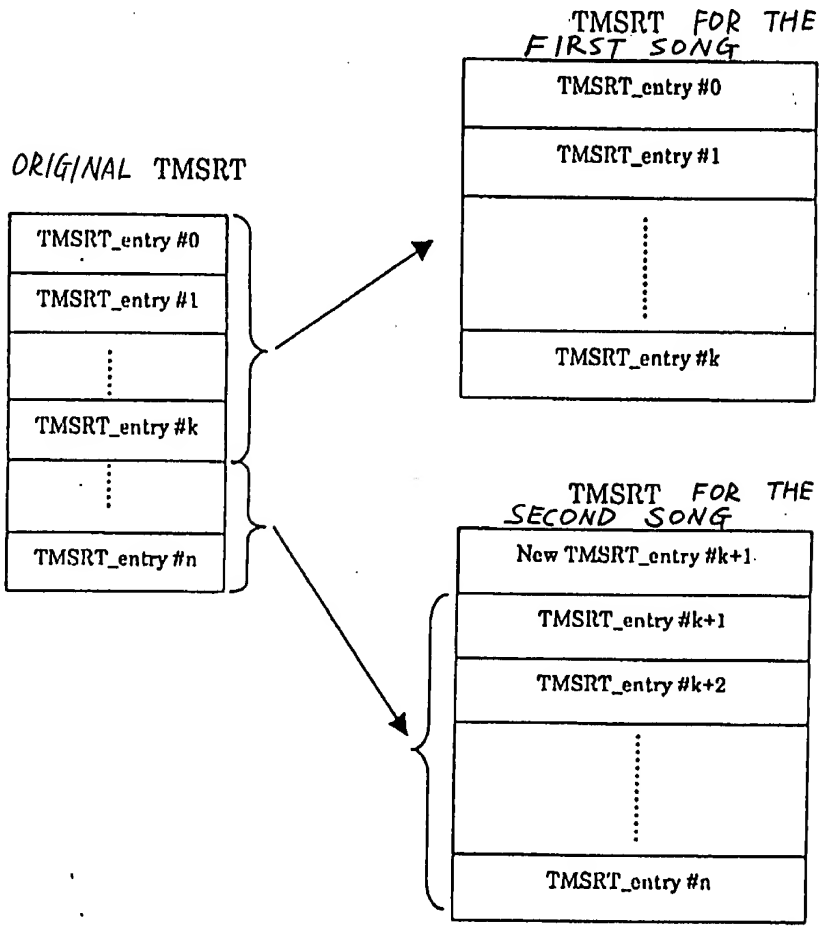


FIG. 25

ORIGINAL BIT

| Data_Offset | 0 |
|---|---|
| SZ_DATA | 52428 |
| TMSRTE_Ns | n+1 |
| TMSRT_Offset | 0 |
| FNs_1st_TMSRTE | 0 |
| FNs_Last_TMSRTE | 50 |
| FNs_Middle_TMSRTE | 96 |

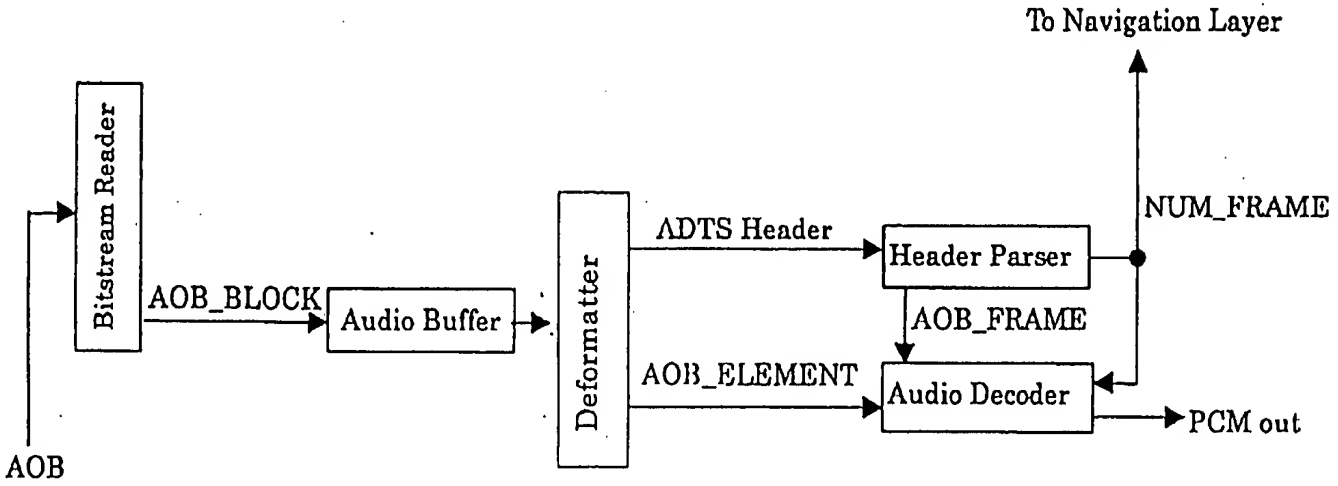
BIT FOR THE FIRST SONG

| Data_Offset | 0 |
|---|---|
| SZ_DATA | Q |
| TMSRTE_Ns | k+1 |
| TMSRT_Offset | 0 |
| FNs_1st_TMSRTE | 0 |
| FNs_Last_TMSRTE | p |
| FNs_Middle_TMSRTE | 96 |

BIT FOR THE SECOND SONG

| Data_Offset | Q |
|---|---|
| SZ_DATA | 52428-Q |
| TMSRTE_Ns | n-k |
| TMSRT_Offset | −TMSRT_entry #0 + Q |
| FNs_1st_TMSRTE | 96-p |
| FNs_Last_TMSRTE | 50 |
| FNs_Middle_TMSRTE | 96 |

FIG. 26




NUM_FRAME : Number Of AOB_FRAME Per AOB_ELEMENT

FIG. 27

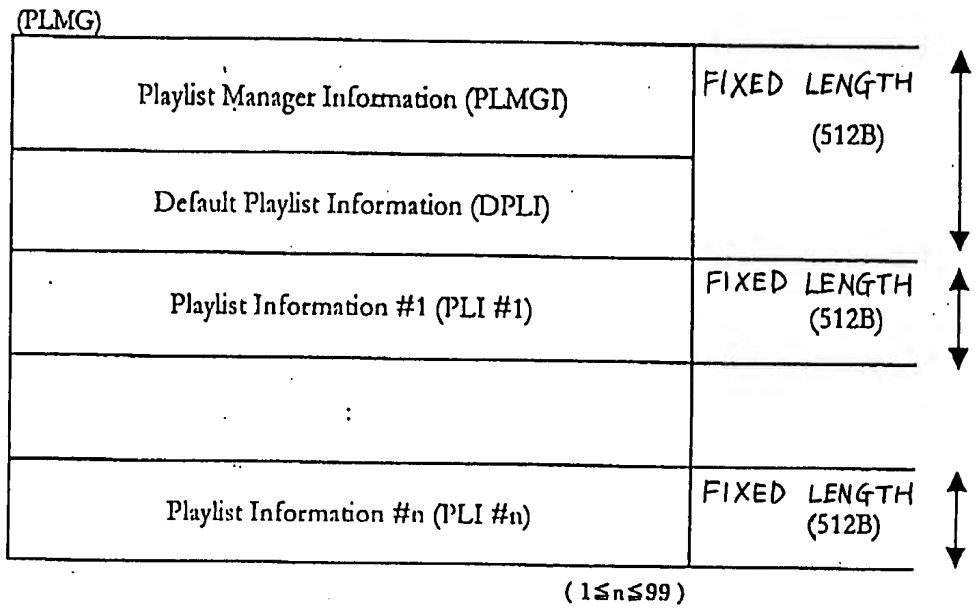


FIG. 28

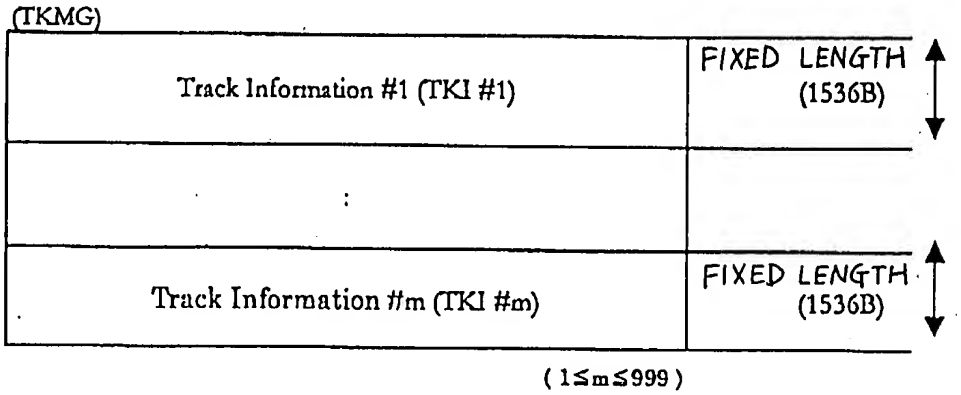


FIG. 29

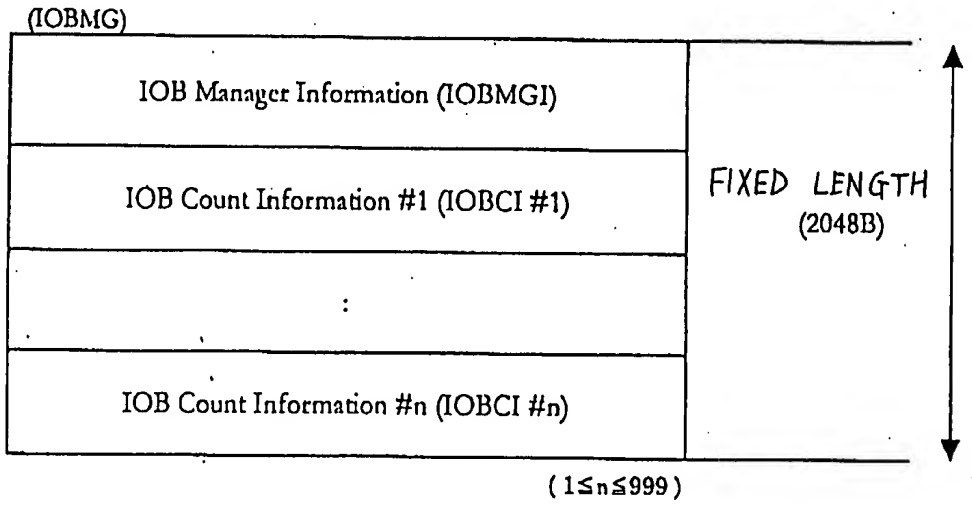


FIG. 30

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | PLMG_ID | ID OF PLMG | 2 BYTES |
| 2 to 3 | reserved | RESERVED | 2 BYTES |
| 4 to 11 | SDA_ID | ID OF SD-Audio | 8 BYTES |
| 12 to 13 | VERN | VERSION NO. OF STANDARD | 2 BYTES |
| 14 to 15 | PLMG_PL_Ns | NO. OF PLAYLISTS | 2 BYTES |
| 16 to 19 | PLMG_AP_PL | PLAYLIST REPRODUCED FIRST | 4 BYTES |
| 20 to 23 | PLMG_RSM_PL | PLAYLIST REPRODUCED LAST | 4 BYTES |
| 24 to 25 | PLMG_APP_ATR | PLMG APPLICATION ATTRIBUTE | 2 BYTES |
| 26 to 27 | PLMG_FCA | Function Code Area | 2 BYTES |
| 28 to 29 | reserved | RESERVED | 2 BYTES |
| 30 to 31 | TKI_Ns | NO. OF TKIs | 2 BYTES |
| 32 to 35 | reserved | RESERVED | 4 BYTES |
| TOTAL | | | 36 BYTES |

FIG. 31

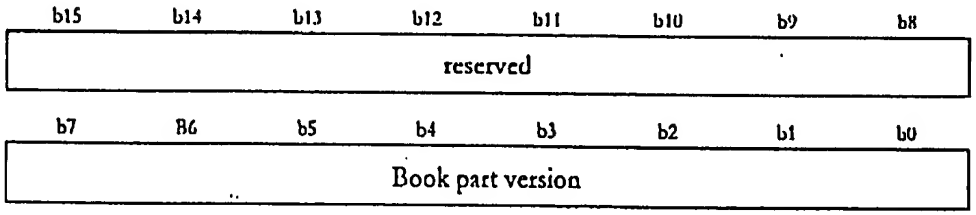


FIG. 32

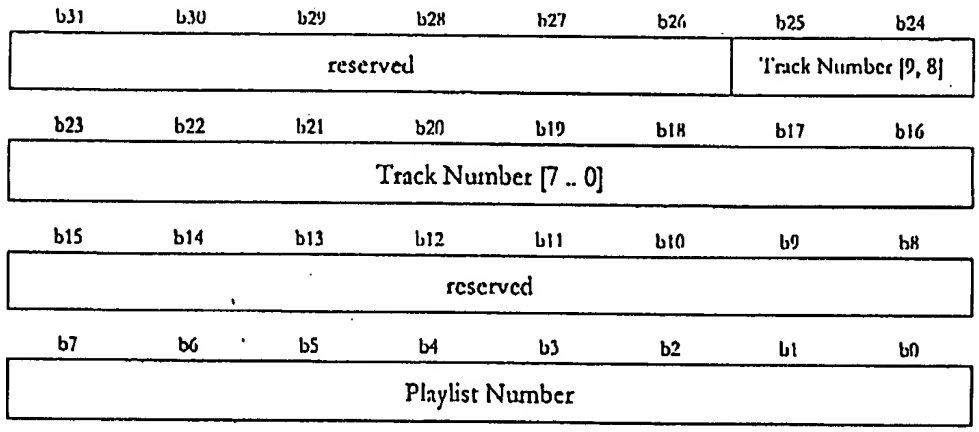


FIG. 33

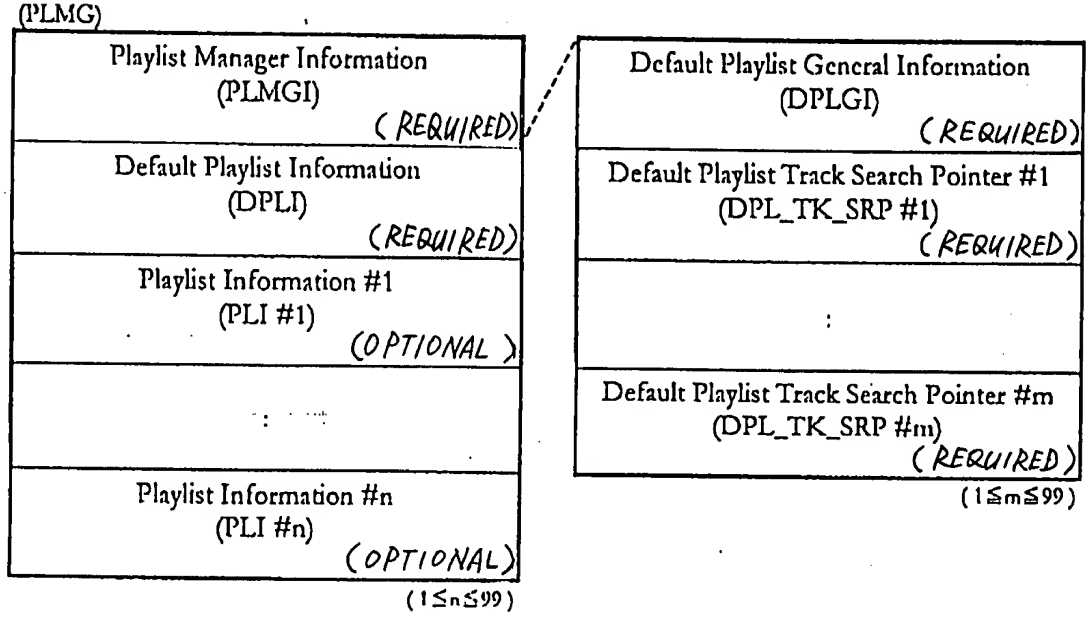


FIG. 34

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | DPLI_ID | ID OF DPLI | 2 BYTES |
| 2 to 3 | reserved | RESERVED | 2 BYTES |
| 4 to 5 | DPLI_TK_Ns | NO. OF SONGS REFFERRED TO BY DEFAULT PLAYLIST | 2 BYTES |
| 6 to 7 | reserved | RESERVED | 2 BYTES |
| 8 to 11 | DPLI_PB_TM | TOTAL REPRODUCTION TIME OF SONGS REFFERRED TO BY DEFAULT PLAYLIST | 4 BYTES |
| 12 to 13 | DPLI_APP_ATR | PLMG APPLICATION ATTRIBUTE | 2 BYTES |
| 14 to 15 | DPLI_FCA | Function Code Area | 2 BYTES |
| 16 to 17 | DPLI_PLTI1_ATR | PLAYLIST TEXT INFORMATION ATTRIBUTE 1 | 2 BYTES |
| 18 to 19 | DPLI_PLTI2_ATR | PLAYLIST TEXT INFORMATION ATTRIBUTE 2 | 2 BYTES |
| 20 to 219 | DPLI_PLTI | TEXT INFORMATION | 200 BYTES |
| 220 to 459 | DPLI_IOB_SRP | DPLI_IOB SEARCH POINTER(4B * 60) | 240 BYTES |
| 460 to 475 | reserved | RESERVED | 16 BYTES |
| TOTAL | | | 476 BYTES |

FIG. 35

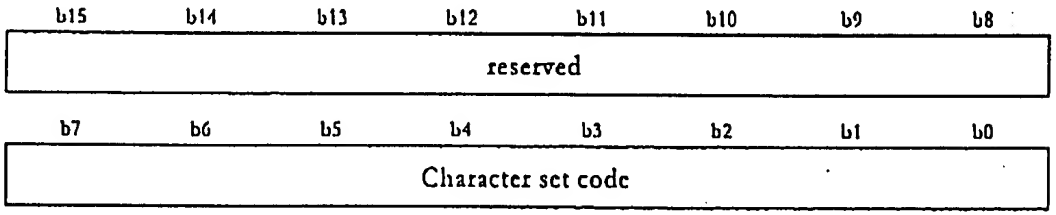


FIG. 36

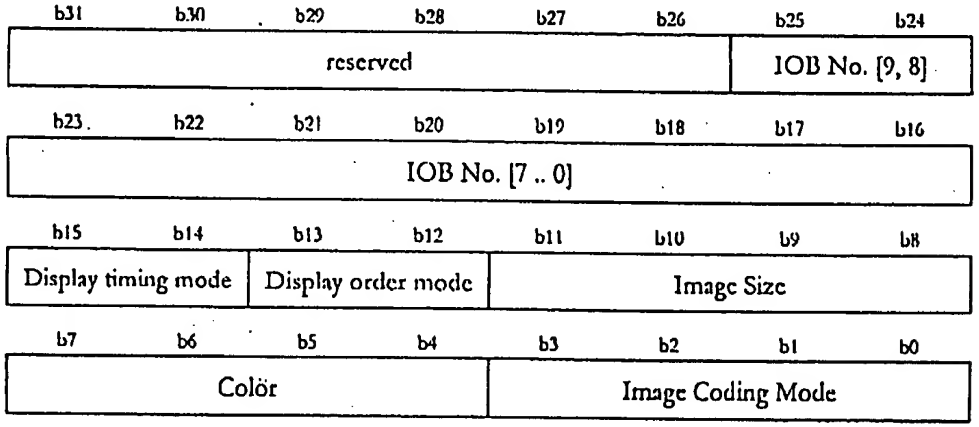


FIG. 37

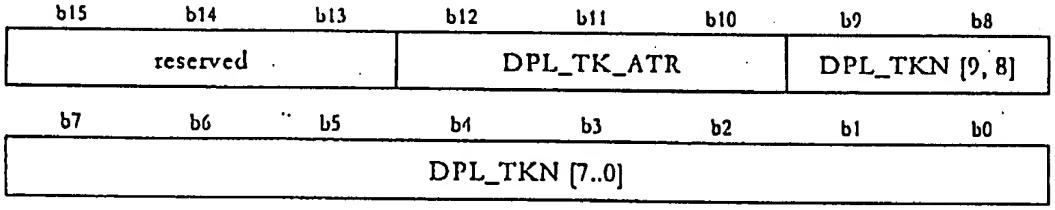


FIG. 38

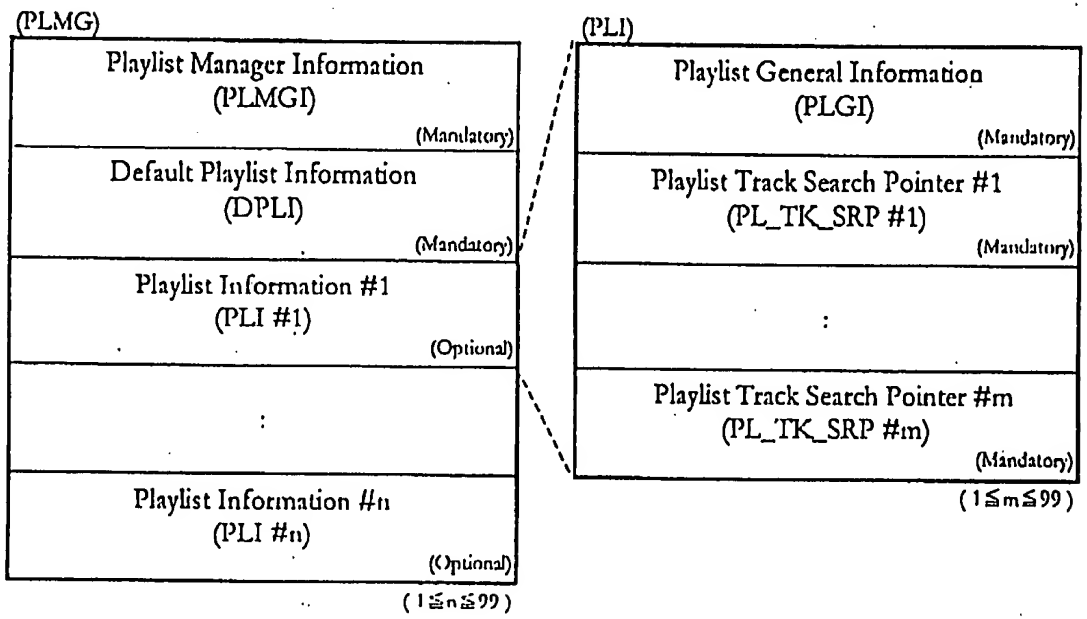


FIG. 39

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | PLI_ID | ID OF PLI | 2 BYTES |
| 2 to 3 | reserved | RESERVED | 2 BYTES |
| 4 to 5 | PLI_TK_Ns | NO. OF SONGS REFFERRED TO BY PLAYLIST | 2 BYTES |
| 6 to 7 | reserved | RESERVED | 2 BYTES |
| 8 to 11 | PLI_PB_TM | TOTAL REPRODUCTION TIME OF SONGS REFFERRED TO BY PLAYLIST | 4 BYTES |
| 12 to 13 | PL_APP_ATR | PLAYLIST APPLICATION ATTRIBUTE | 2 BYTES |
| 14 to 15 | PLI_FCA | Function Code Area | 2 BYTES |
| 16 to 213 | PL_TK_SRPs | PLAYLIST TRACK SEARCH POINTER (2B * 99) | 198 BYTES |
| 214 to 215 | reserved | RESERVED | 2 BYTES |
| 216 to 217 | PLI_PLTI1_ATR | PLAYLIST TEXT ATTRIBUTE INFORMATION 1 | 2 BYTES |
| 218 to 219 | PLI_PLTI2_ATR | PLAYLIST TEXT ATTRIBUTE INFORMATION 2 | 2 BYTES |
| 220 to 419 | PLI_PLTI | PLAYLIST TEXT INFORMATION | 200 BYTES |
| 420 to 499 | PLI_IOB_SRP | PLI_IOB SEARCH POINTER (4B*20) | 80 BYTES |
| 500 to 511 | reserved | RESERVED | 12 BYTES |
| TOTAL | | | 512 BYTES |

FIG. 40

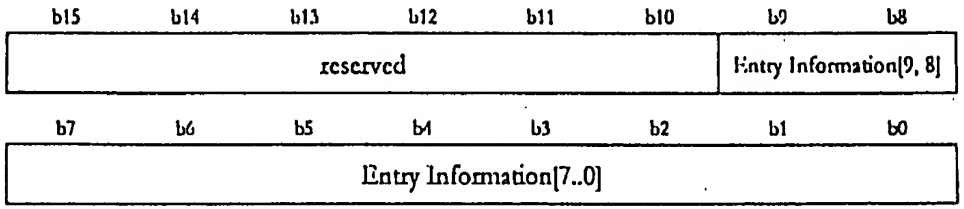


FIG. 41

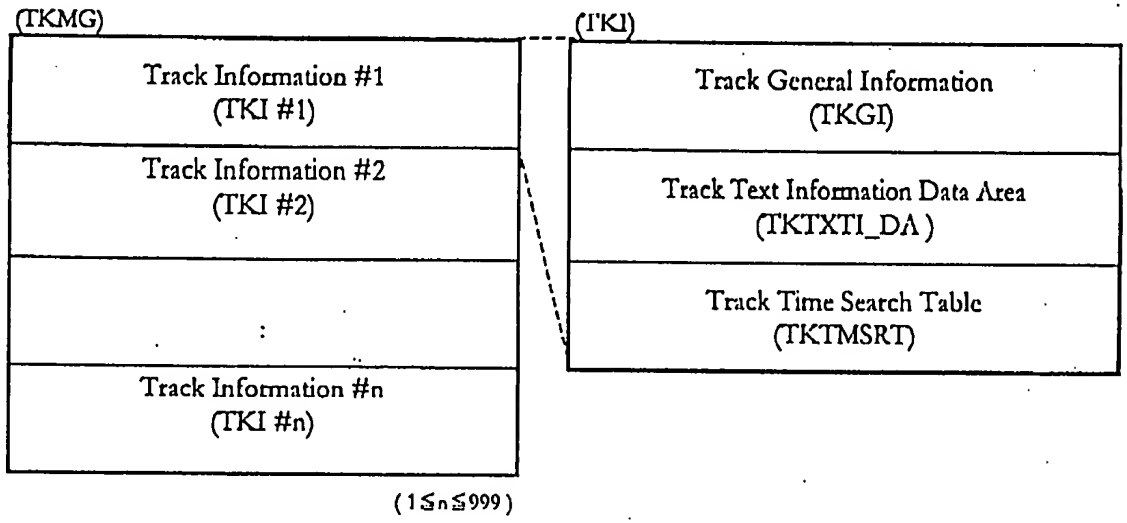


FIG. 42

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | TKI_ID | ID OF TKI | 2 BYTES |
| 2 to 3 | TKI_UI | TKI BLOCK USE ATTRIBUTE | 2 BYTES |
| 4 to 5 | TKIN | TKI NO. | 2 BYTES |
| 6 to 7 | reserved | RESERVED | 2 BYTES |
| 8 to 11 | TKI_SZ | TKI SIZE | 4 BYTES |
| 12 to 13 | TKI_LNK_PTR | LINK POINTER TO NEXT TKI | 2 BYTES |
| 14 to 15 | TKI_BLK_ATR | TKI BLOCK ATTRIBUTE | 2 BYTES |
| 16 to 19 | TKI_PB_TM | REPRODUCTION TIME | 4 BYTES |
| 20 to 23 | TKI_AOB_ATR | TKI AUDIO ATTRIBUTE | 4 BYTES |
| 24 to 27 | reserved | RESERVED | 4 BYTES |
| 28 to 31 | TKI_IOB_ATR | TKI IMAGE ATTRIBUTE | 4 BYTES |
| 32 | reserved | RESERVED FOR COPY MANAGEMENT INFORMATION | 1 BYTE |
| 33 to 35 | reserved | RESERVED | 3 BYTES |
| 36 to 37 | TKI_TI1_ATR | TEXT ATTRIBUTE 1 | 2 BYTES |
| 38 to 39 | TKI_TI2_ATR | TEXT ATTRIBUTE 2 | 2 BYTES |
| 40 to 43 | TKI_TMSRT_SA | TMSRT STARTING POSITION | 4 BYTES |
| 44 to 53 | ISRC | ISRC | 10 BYTES |
| 54 to 55 | reserved | RESERVED | 2 BYTES |
| 56 to 59 | TKI_FCA | Function Code Area | 4 BYTES |
| 60 to 87 | BIT | BLOCK INFORMATION | 28 BYTES |
| 88 to 175 | reserved | RESERVED | 88 BYTES |
| 176 to 255 | TKI_IOB_SRP | TKI_IOB SEARCH POINTER (4B*20) | 80 BYTES |
| TOTAL | | | 256 BYTES |

FIG. 43

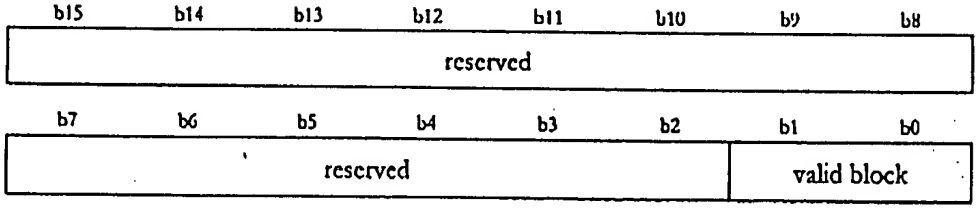


FIG. 44

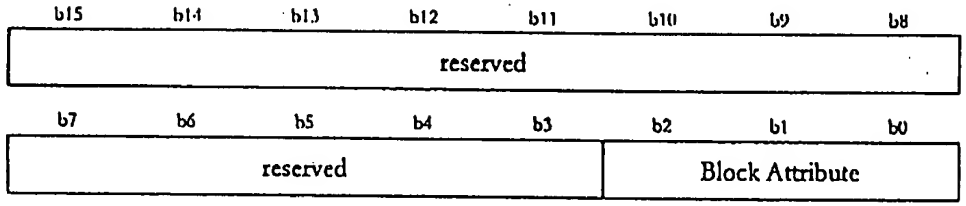


FIG. 45

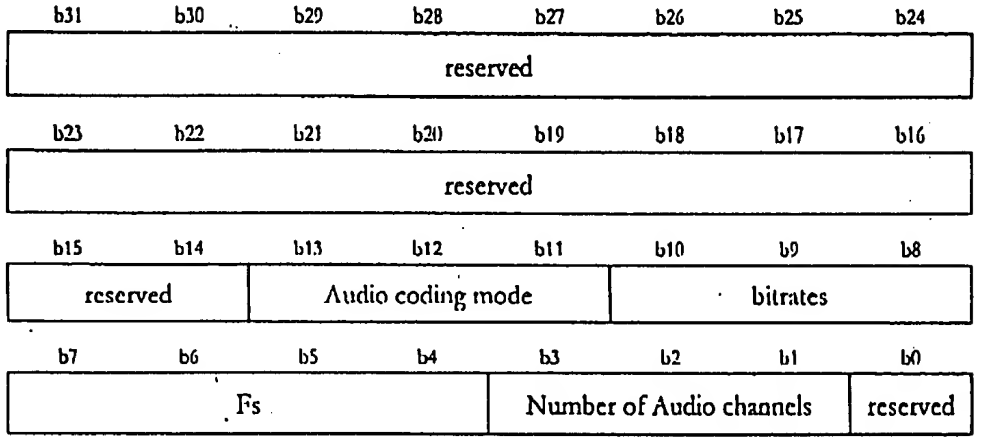


FIG. 46

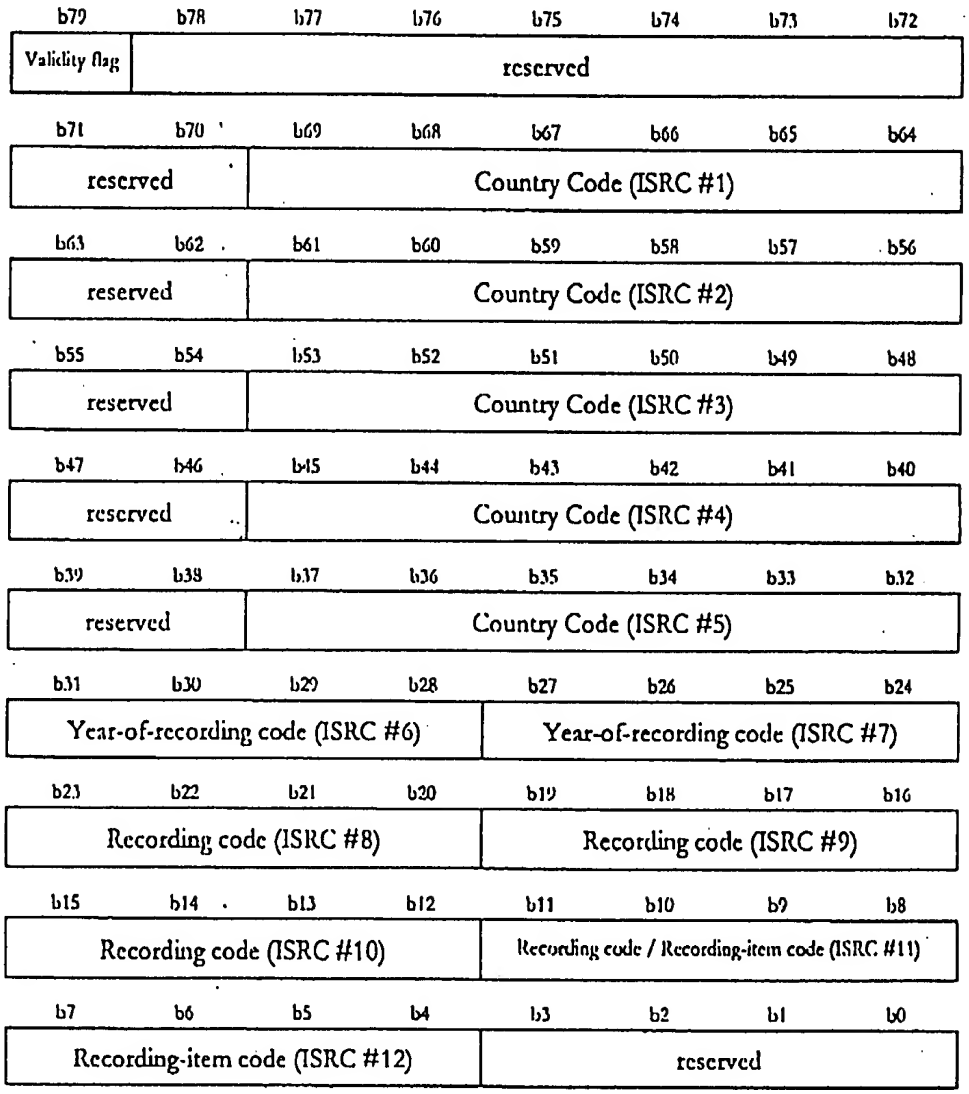


FIG. 47

| SAMPLING FREQUENCY | FNs_Middle_TMSRTE |
|---|---|
| 48kHz | 96 |
| 44.1kHz | 88 |
| 32kHz | 64 |

FIG.48

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 60 to 63 | DATA_OFFSET | AOB_BLOCK STARTING ADDRESS | 4 BYTES |
| 64 to 67 | SZ_DATA | AOB_BLOCK DATA LENGTH | 4 BYTES |
| 68 to 71 | TMSRTE_Ns | NO. OF TMSRT_entry | 4 BYTES |
| 72 to 75 | TMSRT_OFFSET | DATA OFFSET | 4 BYTES |
| 76 to 79 | FNs_1"_TMSRTE | FRAME OFFSET | 4 BYTES |
| 80 to 83 | FNs_Last_TMSRTE | NO. OF AOB_FRAMEs IN LAST AOB_ ELEMENT | 4 BYTES |
| 84 to 87 | FNs_Middle_TMRTE | NO. OF AOB_FRAMEs IN AOB_ ELEMENTC | 4 BYTES |
| TOTAL | | | 28 BYTES |

FIG. 49

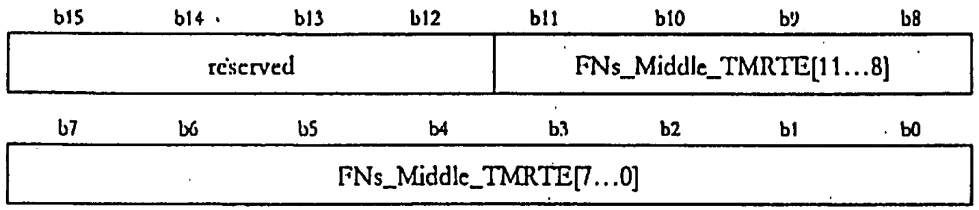


FIG. 50

| TEXT NAME | TAG VALUE | CONTENTS |
|---|---|---|
| TKTXTI_TTL | 01h | TITLE NAME |
| TKTXTI_ART | 02h | ARTIST NAME |
| TKTXTI_ABM | 03h | ALBUM NAME |
| TKTXTI_SW | 04h | SONG WRITER |
| TKTXTI_CMP | 05h | COMPOSER |
| TKTXTI_ARR | 06h | ARRANGER |
| TKTXTI_PRD | 07h | PRODUCER |
| TKTXTI_RCD | 08h | RECORD COMPANY |
| TKTXTI_MSS | 09h | ARTIST'S MESSAGE |
| TKTXTI_UCM | 0Ah | USER'S COMMENT |
| TKTXTI_PCM | 0Bh | PROVIDER'S COMMENT |
| TKTXTI_CRD | 0Ch | YEAR, MONTH, DAY |
| TKTXTI_GNR | 0Dh | GENRE |
| TKTXTI_URL | 0Eh | URL |
| TKTXTI_FR1 | 0Fh | Free 1 |
| TKTXTI_FR2 | 10h | Free 2 |
| TKTXTI_FR3 | 11h | Free 3 |
| TKTXTI_FR4 | 12h | Free 4 |
| TKTXTI_FR5 | 13h | Free 5 |
| TKTXTI_FR6 | 14h | Free 6 |

FIG. 51

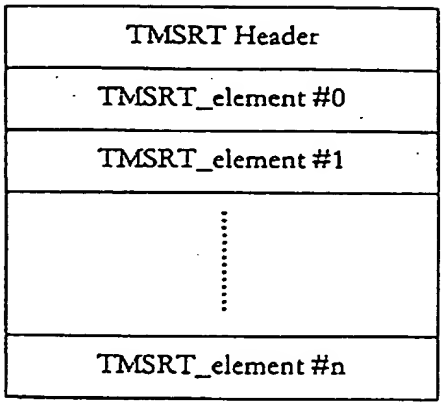


FIG. 52

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | TMSRT ID | ID OF TMSRT | 2 BYTES |
| 2 to 3 | reserved | RESERVED | 2 BYTES |
| 4 to 7 | Total TMSRT_entry Number | TOTAL NO. OF TMSRT_entry | 4 BYTES |
| TOTAL | | | 8 BYTES |

FIG. 53

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 3 | TMSRT_ENT .. | Head Address of AOB_ELEMENT | 4 BYTES |
| TOTAL | | | 4 BYTES |

FIG. 54

(IN THE ORDER OF DESCRIPTION)

| RELATIVE BYTE POSITION | FIELD NAME | CONTENTS | NO. OF BYTES |
|---|---|---|---|
| 0 to 1 | IOBMGI_ID | | 2 BYTES |
| 2 to 3 | reserved | | 2 BYTES |
| 4 to 5 | IOB_Ns | | 2 BYTES |
| 6 to 7 | reserved | | 2 BYTES |
| | | | 8 BYTES |

FIG. 55

| | CONTENTS | NO. OF BYTES |
|---|---|---|
| (1) IOB_RCN | NO. OF REFERENCES TO THE IOB | 2 BYTES * 999 |

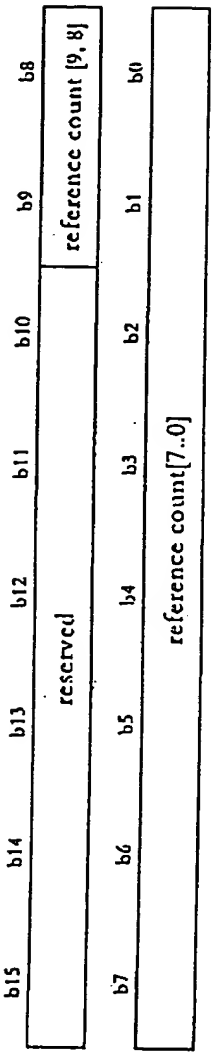



FIG. 56